

银河风云
nROSE 配置手册
IP 服务分册

文档编号：0203_S5300_v5.2_100706

深圳市银河风云网络系统股份有限公司

地址：深圳市科技园北区松坪新西路五号风云大厦

邮编：518057

电话：（0755）83400088

传真：（0755）33630995

客服：800-999-8305

网址：http://www.galaxywind.com

邮箱：customer@galaxywind.com

目录

第1章 IP 寻址和服务

1.1. 概述

Internet 协议（IP）是基于数据包的无连接的协议，它通过计算机网络交换数据。它是所有其它 IP 协议（统称为 IP 协议套件）的基础。作为一种网络层协议，IP 包含寻址和允许数据包路由的控制信息，它处理寻址、分割、拼装及协议信号分解。由于是一个无连接的协议，所以它不需要预定义一个与逻辑网络连接的关联路径。

传输控制协议（TCP）建立在 IP 层基础上。TCP 是一个面向连接的协议，它指定数据格式以及对传输数据的确认。TCP 还指定计算机确保数据正确到达的过程。TCP 允许在一个系统上有多个应用并行通信，因为它能处理应用程序间所有的传输信号分解。

IP 数据包由三层交换机接收，是进行网上（包括因特网）传送的基本单位。每个数据包都包括源地址和目的地址、控制信息以及传向主机层或来自主机层的真实数据。IP 数据包的包头格式见图 1-1。当三层交换机接收到一份 IP 数据包时，便检查它的包头，查找目的网络号和路由表用来确定到达最终目的地址的最佳路由。这样尽管 IP 没有关于数据路径用法的控制，但当一个资源不可用时，它为数据包重选路由。



版本	包头长度	服务类型	总长	识别码	标志	标志偏移量	生存时间	协议	头部校验和	源IP地址	目的IP地址	IP选项	填充	数据
(4位)	(4位)	(1)	(2)	(2)	(3位)	(13位)	(1)	(1)	(2)	(4)	(4)	(可变)	(可变)	(65 500位)

图 1-1 IP 包头格式

IP 连接由 IP 地址控制。IP 地址是指分配给连接在 Internet 上的主机接口的一个 32 位比特地址，唯一标识网络上的一个节点，表示 IP 数据包将要发送到的位置。IP 地址是用来沿网络传递数据包的，就像邮政局向全国和全世界按邮政编码顺次递送信件和包裹一样。三层交换机在进行 IP 报文转发时，对 IP 报文源地址为全零或全 1 的报文进行过滤，对目的地址为 127 系列地址、E 类地址等进行过滤。

所有具有本地地址的数据包都可用 IP 直接传送，而具有外部地址的数据包则根据路由表信息向着目的地址方向传向下一站。

因特网构建在 LAN、FDDI、X.25、ATM 等各种不同类型的底层网络上，底层网络将 IP 分组封装起来进行传输。在数据包转发的过程中，三层交换机需要解决怎样通过底层网络（点到点的连接除外）到达 IP 的"下一跳"的问题，即需要进行网络层和链路层间的地址解析。对 IP 地址的解析，可以静态配置，也可以通过针对某链路层协议设计的地址解析机制来动态地完成，如以太网环境中的地址解析协议（Address Resolution Protocol）。

IP 还管理着数据包的大小，如果数据包的大小超过实际网络能传送的最大长度，IP 将会根据网络硬件的处理能力把数据包分成更小的段，这些数据包在最终被传送到目的地址后又被重新组装在一起。

要配置 IP，一个基本和必需的任务是给网络接口分配 IP 地址，以使这些接口有效，并准许在这些接口上使用 IP 与主机通讯。与这个任务相关的事情是决定子网划分和屏蔽 IP 地址。为了配置各种 IP 地址特性，需完成以下任务。第一项任务是必需的，其余任务是可选的：

- 分配 IP 地址到网络接口
- 配置地址解析方法
- 使 IP 路由有效
- 使 IP 桥接有效
- 使综合路由和桥接有效
- 配置一个路由进程
- 配置广播数据包处理
- 配置网络地址翻译（NAT）
- 监视和维护 IP 寻址

1.2. IP 地址配置

1.2.1.IP 地址介绍

所谓 IP 地址就是给每个连接在 Internet 上的主机分配的一个标识。IP 地址的格式有二进制格式和十进制格式。

二进制的 IP 地址共有 32 位。如 10000011,01101011,00000011,00011000，其中的逗号

仅起印刷的分隔作用。

十进制格式是由二进制翻译过去的，即上述的每八位组用一个十进制数表示，目的是让用户和网管人员便于使用和掌握，并以点分隔称为点分法。上例即 131.107.3.24。

IP 地址一般由两部分组成：第一部分为网络号，第二部分为主机号。IP 地址的结构使我们可以在 Internet 上方便地进行寻址。

由于互联网上的每个接口必须有一个唯一的 IP 地址，因此由管理机构有一个管理机构 InterNIC（Internet Network Information Centre）为接入互联网的网络统一分配 IP 地址。InterNIC 只分配网络号。主机号的分配由系统管理员来负责。

为了方便 IP 地址的管理以及组网，Internet 的 IP 地址分成五类，如图 1-2 所示。

BIT	0 1 2 3 4 … 8 … 16 … 24 … 31
A类	0 | 网络号 | 主机号
B类	1 0 | 网络号 | 主机号
C类	1 1 0 | | 网络号 | 主机号
D类	1 1 1 0 | 多播组号
E类	1 1 1 1 0 | 留待后用

图 1-2 IP 地址分类图

目前大量使用中的 IP 地址为 A 类、B 类、C 类地址；D 类地址是一种多播地址，主要是留给 Internet 体系结构委员会 IAB（Internet Architecture Board）使用；E 类地址保留在今后使用。

在使用 IP 地址时，要知道有一些 IP 地址是保留作为特殊用途的，一般不使用。用户可配置的 IP 地址范围如下：

1、A 类地址

- 分配地址范围：0.0.0.0 ~ 127.255.255.255
- 可用地址范围：1.0.0.0 ~ 126.0.0.0
- 全 0 的主机号：表示该 IP 地址就是网络的地址，用于网络路由
- 全 1 的主机号：表示广播地址，即对该网络上所有的主机进行广播

- IP 地址 0.0.0.0：用于启动后不再使用的主机
- 127 系列地址：保留作回路测试，发送到这个地址的分组不会输出到线路上，它们被当作输入分组进行内部处理

2、B 类地址

- 分配地址范围：128.0.0.0 ~ 191.255.255.255
- 可用地址范围：128.0.0.0 ~ 191.254.0.0
- 全 0 的主机号：表示该 IP 地址就是网络的地址，用于网络路由
- 全 1 的主机号：表示广播地址，即对该网络上所有的主机进行广播

3、C 类地址

- 分配地址范围：192.0.0.0 ~ 223.255.255.255
- 可用地址范围：192.0.0.0 ~ 223.255.254.0
- 全 0 的主机号码：表示该 IP 地址就是网络的地址，用于网络路由
- 全 1 的主机号码：表示广播地址，即对该网络上所有的主机进行广播

4、D 类地址

- 分配地址范围：224.0.0.0 ~ 239.255.255.255
- 可用地址范围：无
- D 类地址是组播地址

5、E 类地址：

- 分配地址范围：240.0.0.0 ~ 247.255.255.255
- 可用地址范围：无
- 保留至今后使用

6、其它地址：

- 可用地址：255.255.255.255
- 255.255.255.255 用于局域网广播地址

现在所有的主机支持子网编址，不再把 IP 地址看成由单纯的一个网络号和一个主机号组成，而是把主机号再分成一个子网号和一个主机号，即 IP 地址的屏蔽码。掩码用来识别 IP 地址中网络号的位数。当使用掩码确定一个网络中的子网时，这个掩码就认为是子网掩码。子网划分缩减了路由表的规模。

子网的划分纯属本单位内部的事，对外部是透明的。从外部看这个单位只有一个网络号，只有当外面的报文进入到本单位范围后，本单位的三层交换机根据子网号再进行选路找到目的主机。

所以，在 InterNIC 获得某类 IP 网络号后，就由当地的系统管理员来进行分配，由管理员根据实际网络情况来决定是否建立子网，以及分配多少比特给子网号和主机号。

例如，这里有一个 B 类网络地址（140.252），在剩下的 16bit 中，8bit 用于子网号，8bit 用于主机号，格式如图 1-3 所示。这样就允许有 254 个子网，每个子网可以有 254 台主机。

	16 BIT	8BIT	8 BIT
B类	网络号=140.252	子网号	主机号

图 1-3 网络地址分配图

子网号还说明两个 IP 地址是否属于一个网段。如果属于同一网段，这两个地址间的信息交换就不通过三层交换机或网桥。如果不属同一网段，也就是子网号不同，两个地址的通讯就要通过三层交换机。

IP 地址还有一些重要的特点：

- 当一个主机同时连接到两个网络上时，例如用作三层交换机的主机，该主机就必须同时具有两个 IP 地址。根据其网络号不同，以对应两个不同的网络。这种主机称为多地址主机 multihomed host。
- 用转发器或网桥连接起来的若干个局域网仍为一个网络，因此这些局域网都具有同样的网络号。
- 若不对子网进行划分，则其子网掩码即为默认值。此时子网掩码中 1 的长度就是网络号码的长度，因此对于 A、B 和 C 类的 IP 地址，其对应子网掩码的默认值分别为 255.0.0.0、255.255.0.0 和 255.255.255.0。
- 三层交换机可用来连接多个子网，可具有多个子网的 IP 地址。
- 在 IP 地址中，所有分配到网络号的网络不管是小的局域网还是很大的广域

网都是平等的。

1.2.2.配置接口 IP 地址

Tritium 三层交换机支持一个接口配置多个 IP 地址，其中一个为主 IP 地址，其余为从 IP 地址。如果在网段上的任何三层交换机使用一个从 IP 地址，那么相同网段上的所有其他三层交换机也必须在相同网段或子网上使用从 IP 地址。

通过配置从 IP 地址使同一接口能位于不同的子网上，从而产生以同一接口为输出端口的网络路由，这样通过同一接口实现与多个子网相连。以下是从 IP 地址最普通的应用场合：

对一个特定的网段也许没有足够的主机地址。例如你的子网允许每一个逻辑子网多达 254 个主机，但是在一个物理子网中你必须有 300 个主机地址。在三层交换机或访问服务器上使用辅助 IP 地址，允许使用一个物理子网的两个逻辑子网。

过去有许多网络使用二级网桥，不是子网。辅助地址的明智使用可以帮助转换到一个子网，即以三层交换机为基础的网络。在一个老的、网桥段上的三层交换机，可以容易地在该段上建立多个子网。

单个网上的两个子网在别的方式下可以被另一个网公开。你可以从子网中建立单个网络，这些子网可以通过使用从 IP 地址被另一个网络从物理上分开。在这些情况下，第一个网在第二个网的顶部被实际扩展或迭加。注意，一个子网不能同时在多于一个活动接口上出现。

1.2.2.1. 配置接口主 IP 地址

表 1–1 配置接口主 IP 地址

命令	**命令模式**	**功能说明**
ip address *IPAddr netmask*	接口配置模式	配置接口主 IP 地址。
show running-config	特权用户模式	显示接口 IP 地址配置信息。

如果三层交换机以太网口 IP 地址是 192.168.1.1，掩码是 255.255.255.0，将 IP 地址与掩码相‘与’后可知三层交换机以太网接口所在网段的地址为 192.168.1.0。

在配置接口主 IP 地址时，需要注意以下问题：

- 一个接口只能配置一个主 IP 地址。配置主 IP 地址时，如果接口上已经配置了 IP 地址，则原主 IP 地址将被新的 IP 地址替代。

- 删除接口 IP 地址时，若未指定具体的 IP 地址和掩码，将删除该接口上所有的 IP 地址（包括所有从 IP 地址），否则必须先删除从 IP 地址，才能删除主 IP 地址。
- 只有 Loopback 接口才能配置 32 位掩码，其它接口最多只能配置 30 位掩码。

1.2.2.2. 配置接口从 IP 地址

除主 IP 地址外，一个接口上还可配置多个从 IP 地址。配置从 IP 地址的主要目的是使同一接口能位于不同的子网上，从而产生以同一接口为输出端口的不同网络的路由，达到同一接口与多个子网相连的目的。

需要注意的是：同一接口的 IP 从地址之间，或同一接口的 IP 从地址和主地址之间可以工作在同一网段上。

表 1–2 配置接口从 IP 地址

命令	命令模式	功能说明
ip address *IPAddr netmask* **secondary**	接口配置模式	配置接口从 IP 地址。
show running-config	特权用户模式	显示接口 IP 地址配置信息。

只有删除了所有从地址后，才允许删除主地址。在有从地址时，不能删除主地址。而且，如果需删除从地址时，必须加上后缀 **secondary**。若该接口上未配主 IP 地址，则不能配置从 IP 地址。

1.2.2.3. 删除接口 IP 地址

表 1–3 删除接口 IP 地址

命令	命令模式	功能说明
no ip address [*IPAddr*]	接口配置模式	删除接口的 IP 地址。
no ip address *IPAddr netmask* **secondary**	接口配置模式	删除接口从 IP 地址。

接口 IP 地址的删除将对与该接口相关联的其它模块功能造成影响，但不会影响相应的配置。

- 直接影响：影响该网段的报文收发。

- 间接影响：导致关联静态路由的删除。因为静态路由的下一跳与接口地址相关。此后若下一跳变为可达，由于配置仍然存在，该静态路由将自动添加。
- 间接影响：导致关联 ARP 表项的删除或不可用（动态 ARP 直接删除，静态 ARP 关联的接口被置空，该 ARP 表项变为不可用）。
- 间接影响：导致路由协议 RIP、OSPF 关联接口删除，但配置未删除，一旦接口重新配置生效，可重新激活该 RIP 或 OSPF 接口。
- 间接影响：导致路由协议 BGP 邻居关系的断开。

【示例】接口 IP 地址删除操作的影响。

例 1：接口 IP 删除前后系统中 ARP 表项的改变。

！查看接口 IP 删除前的 ARP 表项

Tritium# show arp

```
Protocol   Address Age(min)      HardwareAddr    Type      Interface
Internet 200.26.242.2    19   0005.5d0f.2f60    ARPA    FastEthernet2/24
Internet 200.26.242.1    -    0000.0000.0000    ARPA    FastEthernet2/24
Internet 200.26.22.177   17   0005.5d0f.2f60    ARPA    FastEthernet2/24
Internet 200.26.232.1    -    0000.0000.0013    ARPA    FastEthernet2/23
Internet 200.26.12. 1    -    0000.0000.0011    ARPA    FastEthernet2/1
Internet 192.168.0.177   13   0005.5d0f.2f60    ARPA    FastEthernet2/2
```

！删除接口 fastethernet 2/24 的 IP 地址

Tritium(config)# interface fastethernet *2/24*

Tritium(config-if)# no ip address *200.26.242.161 255.255.255.0*

！查看删除接口 IP 后的 ARP 表项

Tritium# show arp

```
Protocol   Address Age(min)      HardwareAddr    Type      Interface
Internet 200.26.22.177   17   0005.5d0f.2f60    ARPA
FastEthernet2/24
Internet 200.26.232.1    -    0000.0000.0013    ARPA
FastEthernet2/23
Internet 200.26.12. 1    -    0000.0000.0011    ARPA    FastEthernet2/1
Internet 192.168.0.177   13   0005.5d0f.2f60    ARPA    FastEthernet2/2
```

Internet 200.26.242. 1 - 0000.0000.0000 ARPA NULL

例 2：接口 IP 删除前后系统中 RIP 配置的改变。

！查看接口 IP 删除前的 RIP 配置内容

Tritium# show ip rip

RIP tasks are running.
Default version control: send version 2, receive version 2
network mask Send Recv SH/PR
Key-chain
8 200.26.52.161 255.255.255.0 1/2 1/2 T/F
5 200.26.22.161 255.255.255.0 1/2 1/2 T/F
27 200.26.242.161 255.255.255.0 1/2 1/2 T/F
Total 3(=3) routes.

！删除接口 fastethernet 2/24 的 IP 地址

Tritium(config)# interface fastethernet *2/24*

Tritium(config-if)# no ip address *200.26.242.161 255.255.255.0*

！查看删除接口 IP 后系统中的 RIP 配置

Tritium# show ip rip

RIP tasks are running.
Default version control: send version 2, receive version 2
network mask Send Recv SH/PR
Key-chain
8 200.26.52.161 255.255.255.0 1/2 1/2 T/F
5 200.26.22.161 255.255.255.0 1/2 1/2 T/F

1.2.2.4. 配置接口转发子网广播

表 **1–4** 配置接口转发子网广播

命令	命令模式	功能说明
ip directed-broadcast	接口配置模式	允许接口转发子网广播。
no ip directed-broadcast	接口配置模式	禁止接口转发子网广播。

该命令只能在 L3 使用。接口默认为允许接口转发子网广播。

1.2.3.配置接口 IP 地址示例

例 1：为交换机的快速以太网接口（槽位号为 2、接口号为 1）配置 IP 地址，要求主 IP 地址为 172.26.12.1，从地址为 172.26.22.1。

Tritium(config)# interface fastethernet *2/1*

Tritium(config-if)# no switchport

Tritium(config-if)# ip address *172.26.12.1 255.255.255.0*

Tritium(config-if)# ip address *172.26.22.1 255.255.255.0* **secondary**

例 2：交换机的快速以太网接口（槽位号为 2、接口号为 1）已经配置了主 IP 地址为 172.26.12.1，从地址为 172.26.22.1。要求删除从 IP 地址 172.26.22.1。

Tritium(config)# interface fastethernet *2/1*

Tritium(config-if)# no switchport

Tritium(config-if)# no ip address *172.26.22.1 255.255.255.0* **secondary**

例 3：交换机的快速以太网接口（槽位号为 2，接口号为 1）已经配置了 IP 地址为 172.26.12.1，要求删除主 IP 地址 172.26.12.1。

Tritium(config)# interface fastethernet *2/1*

Tritium(config-if)# no switchport

Tritium(config-if)# no ip address *172.26.12.1 255.255.255.0*

例 4：为交换机的 POS 接口（槽位号为 1、接口号为 1）配置 IP 地址，要求主 IP 地址为 172.26.11.1，从地址为 172.26.21.1。

Tritium(config)# interface pos *2/1*

Tritium(config-if)# ip address *172.26.11.1 255.255.255.0*

Tritium(config-if)# ip address *172.26.21.1 255.255.255.0* secondary

例 5：配置 VLAN 100 的接口 IP 地址，要求主 IP 地址为 172.26.11.1，从地址为 172.26.21.1。（VLAN 创建略）

Tritium(config)# interface *vlan 100*

Tritium(config-if)# ip address *172.26.11.1 255.255.255.0*

Tritium(config-if)# ip address *172.26.21.1 255.255.255.0* **secondary**

1.2.4.IP 地址配置排错

三层交换机是网络互连设备，因而在给接口配置 IP 地址时，我们必须明白组网需求和子网的划分，一般遵循如下原则：

- 成功配置：在需要配置接口的接口配置模式下成功配置 IP 地址；
- 链路状态指示正确：接口的管理状态和链路状态都应为 UP；
- 以太网口主 IP 地址必须与该以太网口所连的局域网在同一网段；

广域网两端的三层交换机的接口的 IP 地址必须在同一网段。

故障之一：从三层交换机 Ping 局域网中某一主机不通。

故障排除：

- 首先检查该以太网口和局域网中主机的 IP 地址配置是否位于同一网段；
- 如果配置正确就可打开 ARP 调试开关，查看三层交换机是否正确地发送和接收 ARP 报文，如果只有发送没有接收到 ARP 报文，则以太网物理层可能有问题。

故障之二：接口封装 PPP 等协议时，链路层协议状态没有变为 UP。

故障排除：

- 检查该接口 IP 地址与对端是否在同一网段上。

1.3. ARP 配置

1.3.1.概述

在 IP 中，一个设备既可以有本地地址（在局部段或局域网上唯一地标识该设备），也可以有一个网络地址（标识该设备所在的那个网络）。本地地址称作数据链路地址更合适一些，因为它包含在数据包头的数据链路层（OSI 模型的第二层），由数据链路设备读取（例如：网桥和所有设备接口）。从更技术性的角度看，将更像 MAC 地址这样的本地地址，因为在数据链路层里的介质访问控制（MAC）子层为该层处理地址。

例如，为了在以太网上与一个设备通讯，必须首先决定设备的 48 位 MAC 或局部数据链路地址。从一个 IP 地址决定局部数据链路地址的过程称作地址解析。从局部数据链路地址决

定 IP 地址的过程称作反地址解析。

ARP（Address Resolution Protocol）协议主要用于从 IP 地址到 MAC 地址的解析。一般情况下 ARP 动态执行，并自动寻求 IP 地址到 MAC 地址的解析，无需管理员的介入。ARP 缓存表里的 IP 地址与 MAC 地址是相对应的，如：

ip地址	mac地址
192.168.1.1	00－aa－00－62－c6－09
192.168.1.2	00－aa－00－62－c5－03
192.168.1.3	03－aa－01－75－c3－06
······	······

图 1-4 IP 地址与 MAC 地址相对应

ARP 缓存表采用了老化机制，在一段时间内如果表中某一表项没有被使用就会被删除，这样可以大大减少 ARP 缓存表的长度，加快查询速度。

以主机 A（192.168.1.5）向主机 B（192.168.1.1）发送数据为例。当发送数据时，主机 A 会在自己的 ARP 缓存表中寻找是否有目标 IP 地址。如果找到了，也就知道了目标 MAC 地址，直接把目标 MAC 地址写入以太帧里面发送就可以了；如果在 ARP 缓存表中没有找到相对应的 IP 地址，主机 A 就会在网络上发送一个目标 MAC 地址是“ff.ff.ff.ff.ff.ff”的广播帧，以向同一网段内的所有主机发出这样的询问：“192.168.1.1 的 MAC 地址是什么？”网络上其他主机并不响应 ARP 询问，只有主机 B 接收到这个帧时，才向主机 A 做出这样的回应：“192.168.1.1 的 MAC 地址是 00-aa-00-62-c6-09”。这样，主机 A 就知道了主机 B 的 MAC 地址，它就可以向主机 B 发送信息了。同时它还更新了自己的 ARP 缓存表，下次再向主机 B 发送信息时，直接从 ARP 缓存表里查找就可以了。

配置地址解析包括：

- 配置静态 ARP 缓存
- 配置 ARP Proxy

1.3.2.配置静态 ARP 缓存

一般情况下，ARP 表由 ARP 协议动态维护，无需管理人员的介入。在需要添加静态 ARP 表项时，才用 ARP 手工配置命令对 ARP 表进行操作。静态 ARP 表项在主机正常工作时间一

直有效，而动态 ARP 表项的有效时间缺省设置为 7200 秒。

ARP 表缺省为空，由 ARP 协议动态获取地址映射。可通过相关命令显示当前 ARP 表中的表项。

表 1–5 配置静态 ARP 缓存命令列表

命令	命令模式	功能说明
arp *IPAddr PHYAddr*	全局配置模式	手工添加静态 ARP 映射项。
no arp *IPAddr*	全局配置模式	手工删除静态 ARP 映射项。
arp timeout *secs*	全局配置模式	设置 ARP 表项存在的时间，超过这个时间表项将被删除。
no arp timeout	全局配置模式	恢复 ARP 表项存在时间的缺省值。
clear arp-cache	特权用户模式	清除 ARP 缓存表。
show running-config	特权用户模式	显示 ARP 表项的配置信息。

【示例】配置静态 ARP **表项**。

例 1：设置 IP 地址 173.18.45.7，对应的 MAC 地址为 00-01-02-97-ae-eb。

Tritium(config)# arp *173.18.45.7 00:01:02:97:ae:eb*

例 2：删除映射表中 IP 地址 173.18.45.7 对应表项。

Tritium(config)# no arp *173.18.45.7*

例 3：设置 ARP 表项超时时间为 9600 秒。

Tritium(config)# arp timeout *9600*

例 4：恢复 ARP 表项超时时间缺省值 7200 秒。

Tritium(config)# no arp timeout

1.3.3.静态 ARP 的监控与维护

可通过 **show** 命令和 **debug** 命令对 ARP 表项进行监控和维护。

表 1–6 监控维护 ARP 命令列表

命令	命令模式	功能说明
show arp	特权用户模式	显示 ARP 映射表。
debug arp	特权用户模式	打开 ARP 调试信息开关。

no debug arp	特权用户模式	关闭 ARP 调试信息开关。

【示例】静态 ARP 的监控与维护

例 1：显示 ARP 映射表。

Tritium# show arp

```
Protocol    Address        Age(min) HardwareAddr    Type       Interface
Internet 192.168.0.177    14     0005.5d0f.2f60    ARPA    FastEthernet1/0
Internet 192.168.0.89       -     00a0.f749.01e8    ARPA    FastEthernet1/0
```

表中有两条映射项，以第一条为例，表示 IP 地址为 192.168.0.177 的 MAC 地址为 00:05:5d:0f:2f:60，该项通过 ARP 协议动态获取（该动态 ARP 表项还有 14 分钟超时）。

例 2：打开 ARP 调试信息开关。

Tritium# debug arp

```
0x3e143b8 (CONSOLE): IP ARP: sent ARP request src 210.26.12.1:
0601.0201.0702, dst 210.26.12.2: ?
0x49cb750 (tNetTask): IP ARP: recvd rep src 210.26.12.2
0000.0000.0018,dst 210.26.12.1 0601.0201.0702, interface FastEthernet1/1
```

以上 debug 信息显示：

- Tritium 三层交换机发送 ARP 请求报文，该报文源地址 210.26.12.1（MAC 为 06:01:02:01:07:02），查询目的地址为 210.26.12.2 的 MAC 地址。
- 210.26.12.1（MAC 为 06:01:02:01:07:02）从快速以太接口 1/1 收到来自 210.26.12.2 的报文，该报文是对 ARP 请求的响应，带上了其需要查询 IP 地址（210.26.12.2）对应的 MAC 地址 00:00:00:00:00:18。

1.3.4.配置代理 ARP

通过配置代理 ARP（在 RFC 1027 中定义）可以帮助不知道路由的主机决定它在其他网络或子网上的介质地址。例如，三层交换机为一个主机接收一个 ARP 请求，而该主机不在与 ARP 请求发送者相同的接口上，如果该三层交换机的所有路径通过其他接口通往那个主机，那么它就会产生一个代理 ARP 应答包，给出它自己的局部数据链路地址，然后发送 ARP 请

求的主机将自己的包发送到三层交换机，由三层交换机将它们转发到目标主机。图 1-5 是代理 ARP 的一个应用过程。

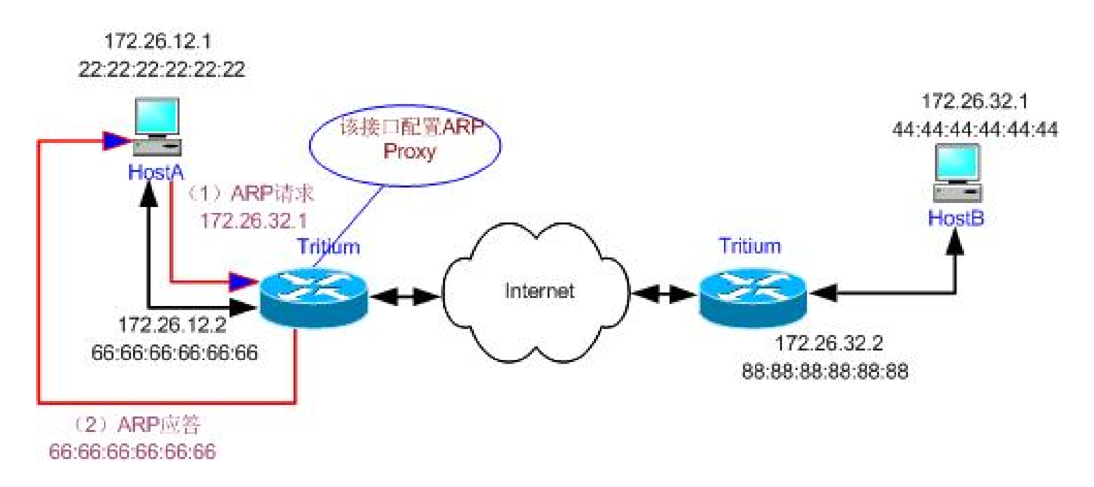


图 1-5 ARP Proxy 应用过程图示

如图所示，HOST A 和 HOST B 分别属于不同的物理网络，IP 地址属于同一 IP 网络 172.26.0.0 的不同子网，没有配置网关，HOST A 向 HOST B 发送 ARP 请求，与 HOST A 网络相连的三层交换机接口已经使能 ARP Proxy 功能，如果存在到 HOST B 的正常路由，则代替 HOST B 回应自己接口的 MAC 地址。HOST A 向 HOST B 发送的 IP 报文都发给了三层交换机，三层交换机对报文作正常的 IP 路由转发，发往 HOST B 的 IP 报文通过网络最终到达 HOST B。反之亦然。

表 1–7 配置 ARP Proxy 命令列表

命令	命令模式	功能说明
ip proxy-arp	接口配置模式	使能接口的 ARP Proxy 功能。
no ip proxy-arp	接口配置模式	禁止接口的 ARP Proxy 功能。

Tritium 三层交换机接口缺省时未开启代理 ARP（Proxy ARP）。

【示例】关闭指定接口的 ARP 代理。

Tritium(config-if)# no ip proxy-arp

1.3.5.ARP Proxy 的监控与维护

三层交换机管理员可通过 **show** 和 **debug** 命令对 ARP Proxy 配置后的情况进行监控和维护。

表 1–8 监控维护 ARP Proxy 命令列表

命令	命令模式	功能说明
show running-config	特权用户模式	显示接口上的 ARP Proxy 配置信息。
debug arp packet	特权用户模式	打开 ARP 调试信息开关，可以观察 ARP Proxy 的运行信息。
no debug arp packet	特权用户模式	关闭 ARP 调试信息开关，可以观察 ARP Proxy 的运行信息。

1.3.6.典型 ARP Proxy 配置实例

1.3.6.1. 组网需求

假设有一家公司在深圳、上海两地各有一个局域网，分别通过交换机与广域网相连，深圳局域网的主机统一使用 172.26.12.0 子网地址，上海局域网的主机统一使用 172.26.32.0 子网地址。两地交换机通过以太接口与局域网相连，通过广域网互连。使用 ARP Proxy 可以使两地的 PC 不需要网关就实现互通，就好像在一个物理网络中。

1.3.6.2. 组网图

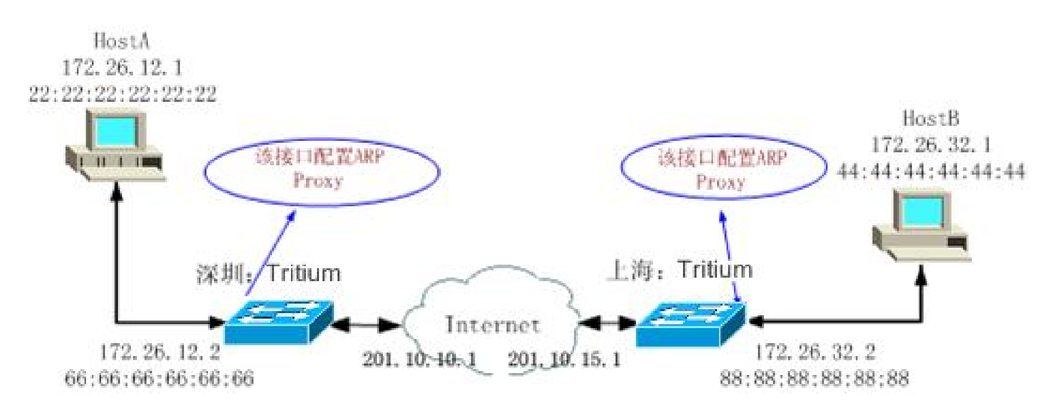


图 1-6 ARP Proxy 配置组网图

1.3.6.3. 配置步骤

配置说明：VLAN 已创建好，下一跳路由 IP 地址分别为 201.10.10.1、201.10.15.1。

（1）配置深圳的交换机

！配置接口 IP 地址后，使能接口的 ARP Proxy，配置到上海的 IP 路由

Tritium(config-vlan-if)# ip address *172.26.12.2 255.255.255.0*

Tritium(config-vlan-if)# ip proxy-arp

Tritium(config-vlan-if)# exit

Tritium(config)# ip route *172.26.32.0 255.255.255.0 201.10.10.1*

（2）配置上海的交换机

！配置接口 IP 地址后，使能接口的 ARP Proxy，配置到深圳的 IP 路由

Tritium(config-vlan-if)# ip address *172.26.32.2 255.255.255.0*

Tritium(config-vlan-if)# ip proxy-arp

Tritium(config-vlan-if)# exit

Tritium(config)# ip route *172.26.12.0 255.255.255.0 201.10.15.1*

1.4. DHCP 配置

1.4.1.DHCP 概述

动态主机配置协议——DHCP 是 Dynamic Host Configuration Protocol 之缩写，它的前身是 BOOTP。BOOTP 原本是用于无磁碟主机连接的网络上面的，网络主机使用 BOOT ROM 而不是磁碟起动并连接上网络，BOOTP 则可以自动地为那些主机设定 TCP/IP 环境。但 BOOTP 有一个缺点，您在设定前须事先获得客户端的硬件地址，而且，与 IP 的对应是静态的。换而言之，BOOTP 非常缺乏“动态性”，若在有限的 IP 资源环境中，BOOTP 的一一对应会造成非常可观的浪费。

DHCP 可以说是 BOOTP 的增强版本，它分为两个部份：一个是服务器端，而另一个是客户端。所有的 IP 网络设定资料都由 DHCP 服务器集中管理，并负责处理客户端 DHCP 要求；而客户端则会使用从服务器分配下来的 IP 环境资料。比较起 BOOTP，DHCP 透过“租约”的概念，有效且动态的分配客户端的 TCP/IP 设定，而且，作为兼容考虑，DHCP 也完全照顾了 BOOTP Client 的需求。

1.4.1.1. DHCP 的分配形式

首先，必须至少有一台 DHCP Server 工作在网络上面，它会监听网络的 DHCP 请求，

并与客户端磋商 TCP/IP 的设定环境。它提供两种 IP 定位方式：

- Automatic Allocation

自动分配，其情形是：一旦 DHCP 客户端第一次成功的从 DHCP 服务器端租用到 IP 地址之后，就永远使用这个地址。

- Dynamic Allocation

动态分配，当 DHCP 第一次从 HDCP 服务器端租用到 IP 地址之后，并非永久的使用该地址，只要租约到期，客户端就得释放（release）这个 IP 地址，以给其它工作站使用。当然，客户端可以比其它主机更优先的延续（renew）租约，或是租用其它的 IP 地址。

动态分配显然比自动分配更加灵活，尤其是当您的实际 IP 地址不足的时候，例如：您是一家 ISP，只能提供 200 个 IP 地址用来给拨接客户，但并不意味着您的客户最多只能有 200 个。因为要知道，您的客户们不可能全部同一时间上网的，除了他们各自的行为习惯的不同，也有可能是电话线路的限制。这样，您就可以将这 200 个地址，轮流的租用给拨接上来的客户使用了。这也是为什么当您查看 IP 地址的时候，会因每次拨接而不同的原因了（除非您申请的是一个固定 IP，通常的 ISP 都可以满足这样的要求，这需要另外收费）。当然，ISP 不一定使用 DHCP 来分配地址，但这个概念和使用 IP Pool 的原理是一样的。

DHCP 除了能动态的设定 IP 地址之外，还可以将一些 IP 保留下来给一些特殊用途的机器使用，它可以按照硬件地址来固定的分配 IP 地址，这样可以给您更大的设计空间。同时，DHCP 还可以帮客户端指定 Router、Netmask、DNS Server、WINS Server 等等项目，您在客户端上面，除了将 DHCP 选项打勾之外，几乎无需做任何的 IP 环境设定。

1.4.1.2. DHCP 的工作原理

对于客户端是否第一次登录网络，DHCP 的工作形式会有所不同。

第一次登录的时候：

1） **寻找 Server。**

当 DHCP 客户端第一次登录网络的时候，也就是客户发现本机上没有任何 IP 资料设定，它会向网络发出一个 DHCPDISCOVER 封包。因为客户端还不知道自己属于哪一个网络，所以封包的来源地址会为 0.0.0.0，而目的地址则为 255.255.255.255，然后再附上 DHCPDISCOVER 的信息，向网络进行广播。

在 Windows 的预设情形下，DHCPDISCOVER 的等待时间预设为 1 秒，也就是当客户端将第一个 DHCPDISCOVER 封包送出去之后，在 1 秒之内没有得到回应的话，就会进行第二次 DHCPDISCOVER 广播。若一直得不到回应的情况下，客户端一共有四次 DHCPDISCOVER 广播（包括第一次在内），除了第一次会等待 1 秒之外，其余三次的等待时间分别是 9、13、16 秒。如果都没有得到 DHCP 服务器的回应，客户端则会显示错误信息，宣告 DHCPDISCOVER 的失败。之后，基于使用者的选择，系统会继续在 5 分钟之后再重复一次 DHCPDISCOVER 的过程。

2）**提供 IP 租用地址。**

当 DHCP 服务器监听到客户端发出的 DHCPDISCOVER 广播后，它会从那些还没有租出的地址范围内，选择最前面的空置 IP，连同其它 TCP/IP 设定，回应给客户端一个 DHCPOFFER 封包。

由于客户端在开始的时候还没有 IP 地址，所以在其 DHCPDISCOVER 封包内会带有其 MAC 地址信息，并且有一个 XID 编号来辨别该封包，DHCP 服务器回应的 DHCPOFFER 封包则会根据这些资料传递给要求租约的客户。根据服务器端的设定，DHCPOFFER 封包会包含一个租约期限的信息。

3）**接受 IP 租约。**

如果客户端收到网络上多台 DHCP 服务器的回应，只会挑选其中一个 DHCPOFFER 而已（通常是最先抵达的那个）。此时客户端会先向向网络发送一个 ARP 封包，查询网络上面有没有其它机器使用该 IP 地址。如果发现该 IP 已经被占用，客户端则会送出一个 DHCPDECLINE 封包给 DHCP 服务器，拒绝接受其 DHCPOFFER，并重新发送 DHCPDISCOVER 信息。 如果发现该 IP 可用，客户端会向网络发送一个 DHCPREQUEST 广播封包，告诉所有 DHCP 服务器它将指定接受哪一台服务器提供的 IP 地址。

事实上，并不是所有 DHCP 客户端都会无条件接受 DHCP 服务器的 Offer，尤其这些主机安装有其它 TCP/IP 相关的客户软件。客户端也可以用 DHCPREQUEST 向服务器提出 DHCP 选择，而这些选择会以不同的号码填写在 DHCP Option Field 里面：

号码	代表意思
01	Sub-net Mask
03	Router Address

06	DNS Server Address
0F	Domain Name
2C	WINS/NBNS Server Address
2E	WINS/NBT Node Type
2F	NetBIOS Scope ID

换一句话说，在 DHCP 服务器上面的设定，未必是客户端全都接受，客户端可以保留自己的一些 TCP/IP 设定。而主动权永远在客户端这边。

4）租约确认。

当 DHCP 服务器接收到客户端的 DHCPREQUEST 之后，会向客户端发出一个 DHCPACK 回应，以确认 IP 租约的正式生效，也就结束了一个完整的 DHCP 工作过程。

如上的工作流程如下图：

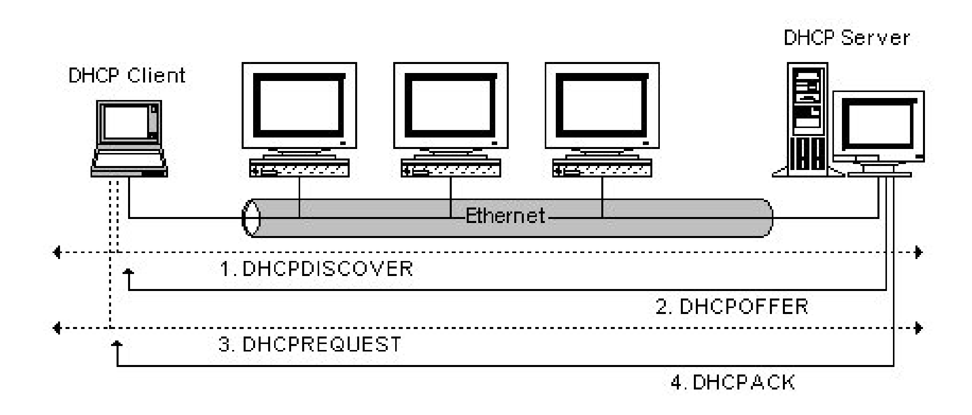


图 1-7 DHCP 工作流程图

第一次登录之后：

一旦 DHCP 客户端成功地从服务器哪里取得 DHCP 租约之后，除非其租约已经失效并且 IP 地址也重新设定为 0.0.0.0，否则就无需再发送 DHCPDISCOVER 信息了，而会直接使用已经租用到的 IP 地址向前面的 DHCP 服务器发出 DHCPREQUEST 信息，DHCP 服务器会尽量让客户端使用原来的 IP 地址，如果没问题的话，直接回应 DHCPACK 来确认则可。如果该地址已经失效或已经被其它机器使用了，服务器则会回应一个 DHCPNACK 封包给客户端，要求其从新执行 DHCPDISCOVER。

至于 IP 的租约期限却是非常考究的，并非如我们租房子那样简单，以 NT 为例子：DHCP 工作站除了在开机的时候发出 DHCPREQUEST 请求之外，在租约期限一半的时候也会发出 DHCPREQUEST，如果此时得不到 DHCP 服务器的确认的话，工作站还可以继续使用该 IP；

然后在剩下的租约期限的再一半的时候（即租约的 75%），还得不到确认的话，那么工作站就不能拥有这个 IP 了。

要是您想退租，可以随时送出 DHCPRELEASE 命令解约，就算您的租约在前一秒钟才获得的。

1.4.1.3. 跨网络的 DHCP 运作

从前面描述的过程中，我们不难发现：DHCPDISCOVER 是以广播方式进行的，其情形只能在同一网络之内进行，因为交换机是不会将广播传送出去的。但如果 DHCP 服务器安设在其它的网络上面呢？由于 DHCP 客户端还没有 IP 环境设定，所以也不知道交换机，而且有些交换机也不会将 DHCP 广播封包传递出去，因此这情形 DHCPDISCOVER 是永远没办法抵达 DHCP 服务器那端的，当然也不会发生 Offer 及其他动作了。要解决这个问题，我们可以用 DHCP 代理服务器（或 DHCP Proxy）来接管客户的 DHCP 请求，然后将此请求传递给真正的 DHCP 服务器，然后将服务器的回复传给客户。这里，DHCP 代理服务器必须自己具有路由能力，且能将双方的封包互传对方。

若不使用 DHCP 代理，您也可以在每一个网络之中安装 DHCP 服务器，但这样的话，一来设备成本会增加，而且，管理上面也比较分散。当然，如果在一个十分大型的网络中，这样的均衡式架构还是可取的。

1.4.1.4. DHCP 封包格式

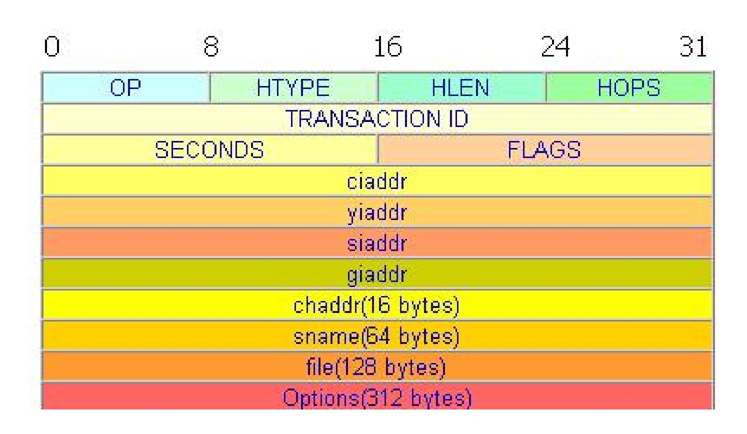


图 1-8 DHCP 封包格式

以下为各栏位的简要说明：

- OP

若是 Client 送给 Server 的封包，设为 1，反向为 2。

- HTYPE

硬件类别，Ethernet 为 1。

- HLEN

硬件地址长度，Ethernet 为 6。

- HOPS

若封包需经过中继代理传送，每站加 1，若在同一网内，为 0。

- TRANSACTION ID

DHCPREQUEST 时产生的数值，以作 DHCPREPLY 时的依据。

- SECONDS

Client 端启动时间（秒）。

- FLAGS

从 0 到 15 共 16bits，最左一个 Bit 为 1 时表示 Server 将以广播方式传送封包给 Client，其余尚未使用。

- ciaddr

要是 Client 端想继续使用之前取得之 IP 地址，则列于这里。

- yiaddr

从 Server 送回 Client 之 DHCPOFFER 与 DHCPACK 封包中，此栏填写分配给 Client 的 IP 地址。

- siaddr

若 Client 需要透过网络开机，从 Server 送出之 DHCPOFFER、DHCPACK、DHCPNACK 封包中，此栏填写开机程序代码所在 Server 之地址。

- giaddr

若需跨网域进行 DHCP 发放，此栏为第一个 Relay Agent 的地址，否则为 0。

- chaddr

Client 之硬件地址。

- sname

Server 之名称字串，以 0x00 结尾。

- file

若 Client 需要透过网络开机，此栏将指出开机程序名称，稍后以 TFTP 传送。

- options

Code	Length	Value

此栏位完全兼容BOOTP，同时扩充了更多选项。其中，DHCP封包可利用编码为0x53的选项来设定封包类别。

项值	类别
1	DHCPDISCOVER
2	DHCPOFFER
3	DHCPREQUEST
4	DHCPDECLINE
5	DHCPACK
6	DHCPNACK
7	DHCPRELEASE

图 1-9 DHCP options 栏位说明图

允许厂商定义选项（Vendor-Specific Area），以提供更多的设定信息（如：Netmask、Gateway、DNS、等等）。其长度可变，同时可携带多个选项，每一选项之第一个 Byte 为信息代码，其后一个 Byte 为该项资料长度，最后为项目内容。

1.4.2.DHCP 配置

1.4.2.1. 三层交换机 DHCP 中继配置

随着网络规模的不断扩大、网络复杂度的不断提高，网络配置也变得越来越复杂。原有的针对静态主机配置的 BOOTP 协议已经越来越不适应人们的需求，在计算机经常移动（如：便携机的使用或无线网络）和实际计算机数量超过可分配的 IP 地址等场合下，BOOTP 协议显得尤为捉襟见肘。

为方便用户快速接入和退出网络、提高 IP 地址资源的利用率以及支持无盘网络工作站等机制，在 BOOTP 协议的基础上，人们制定了动态主机配置协议 DHCP（Dynamic Host Configuration Protocol）。与 BOOTP 协议一样，DHCP 协议也是以客户机/服务器（Client/Server）模式工作的。利用该协议，DHCP 客户机可以向 DHCP 服务器动态地请求配置信息，包括分配的 IP 地址、子网掩码、缺省网关等重要参数，而 DHCP 服务器也可以很方便地为其动态配置这些信息。

早期的 DHCP 协议只适用于 DHCP 客户机和服务器处于同一个子网内的情况，不可以跨网段工作。因此，为进行动态主机配置，需要在所有网段上都设置一个 DHCP 服务器，这显然是不经济的。

DHCP 中继的引入解决了这一难题：它在不同网段间的 DHCP 客户机和服务器之间承担中继服务，可以将 DHCP 协议报文跨网段中继到目的 DHCP 服务器（或客户机），于是许多网络上的 DHCP 客户机可以使用同一个 DHCP 服务器。这样，既节省成本又便于集中管理。

下图是 DHCP 中继的示意图。其工作原理如下：

- 当 DHCP 客户机启动并进行 DHCP 初始化时，它会在本网络广播配置请求报文。
- 如果本网络存在 DHCP 服务器（如上图中右边的以太网），则不需要 DHCP 中继就可以直接进行 DHCP 配置；
- 如果本网络里没有 DHCP 服务器（如上图中左边的以太网），则与本网络相连的、带 DHCP 中继功能的网络设备（图中为三层交换机）在收到该广播报文并适当处理后，将其转发给指定的存在于其它网络上的 DHCP 服务器。
- DHCP 服务器根据客户机提供的必要信息，为其作相应的配置，并再次通过 DHCP 中继将该配置信息发送给客户机，完成对客户机的动态配置。事实上，从开始到最终完成配置，需要多个这样的交互过程。

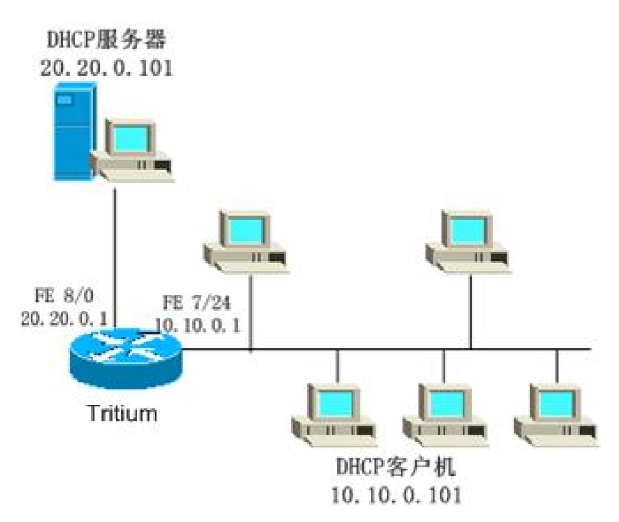


图 1-10 DHCP 中继示意图

实际上，DHCP 中继实现了一种对 DHCP 广播报文的透明传输功能，即把 DHCP 客户机（或服务器）的广播报文透明地传送到其它网段的 DHCP 服务器（或客户机）上。

在运行的三层交换机网络设备中，需要用户配置 IP 中继地址，用来指定 DHCP 服务器地址，只需要执行如下的配置任务：

表 1–9 三层交换机 DHCP 中继配置

命令	命令模式	功能说明
service dhcp	全局配置模式	启动 DHCP 服务。
ip dhcp-server *ipaddress*	全局配置模式	指定 DHCP 服务器地址。

1.4.2.2. 三层交换机充当 DHCP 服务器

Tritium 三层交换机本身可实现 DHCP 服务器功能。在运行的三层交换机网络设备中，用户可根据网络需要进行相应的配置，只需要执行如下的配置任务：

表 1–10 配置 DHCP 基本参数

命令	命令模式	功能说明
service dhcp	全局配置模式	启动 DHCP 服务。
ip dhcp ping packets	全局配置模式	配置在分配给请求地址的客户端 IP 地址之前 Ping Packets 的个数。
ip dhcp ping timeout	全局配置模式	配置在分配给请求地址的客户端 IP 地址之前等待 Ping 回应的时间。
ip dhcp pool *poolname*	全局配置模式	配置 DHCP 地址池名称。

表 1–11 配置 DHCP 地址池范围及参数

命令	命令模式	功能说明
network *net-number* [*mask*]	DHCP 二级配置模式	配置地址池的地址与掩码。
ip dhcp excluded-address *low-address* [*high-address*]	全局配置模式	配置 DHCP 地址池中不分配的 IP 地址。
default-router *address* [*address2… address4*]	DHCP 二级配置模式	配置分配给客户端的默认网关的地址。
dns-server *address* [*address2… address4*]	DHCP 二级配置模式	配置分配给客户端的 DNS 服务器的地址。
domain-name *domain-name*	DHCP 二级配置模式	配置客户端域名。
lease { *days* [*hours*] [*minutes*] }	DHCP 二级配置模式	配置客户端 IP 租用时间。

表 1–12 配置客户端 MAC 地址绑定

命令	命令模式	功能说明
client-identifier *unique-identifier*	DHCP 二级配置模式	配置 Microsoft DHCP Client 指定的唯一标识。
hardware-address	DHCP 二级配置模式	配置 MAC 地址与 IP 地址之间的

hardware-address		绑定。
host *ipaddress* [*mask*]	DHCP 二级配置模式	将特定的 IP（和网络掩码）分配给 DHCP 客户端。
default-router *address* [*address2… address4*]	DHCP 二级配置模式	配置分配给客户端的默认三层交换机的地址。
dns-server *address* [*address2… address4*]	DHCP 二级配置模式	配置分配给客户端的 DNS 服务器的地址。
domain-name *domain-name*	DHCP 二级配置模式	配置客户端域名。
lease { *days* [*hours*] [*minutes*] }	DHCP 二级配置模式	配置客户端 IP 租用时间。

表 1–13 配置 netbios 参数

命令	命令模式	功能说明
netbios-name-server *address* [*address2… address4*]	DHCP 二级配置模式	配置 NetBIOS WINS 服务器的地址。
netbios-node-type *type*	DHCP 二级配置模式	配置 NetBIOS Node 类型。

1.4.3.DHCP 管理及监控

表 1–14 DHCP 管理及监控

命令	命令模式	功能说明
debug ip dhcp event	全局配置模式	打开 DHCP 事件调试开关。
debug ip dhcp packet	全局配置模式	打开 DHCP 报文调试开关。
show dhcp lease	全局配置模式	查看 DHCP 的分配情况。
clear ip dhcp binding { **all** | *address* }	全局配置模式	收回已分配的 IP 地址。

1.4.4.DHCP 配置实例

1.4.4.1. DHCP 中继配置

一、组网需求

DHCP 客户机所在的网段为 10.10.0.0，而 DHCP 服务器所在的网段为 20.20.0.0。需要通过带 DHCP 中继功能的交换机中继 DHCP 报文，使得 DHCP 客户机可以从 DHCP 服务器上申请到 IP 地址等相关配置信息。

DHCP 服务器应当分配一个 10.10.0.0 网段的 IP 地址池，以便将适当的 IP 地址分配给该

网段上的 DHCP 客户机，并且 DHCP 服务器上应当配置有到 10.10.0.0 网段的路由。

二、组网图

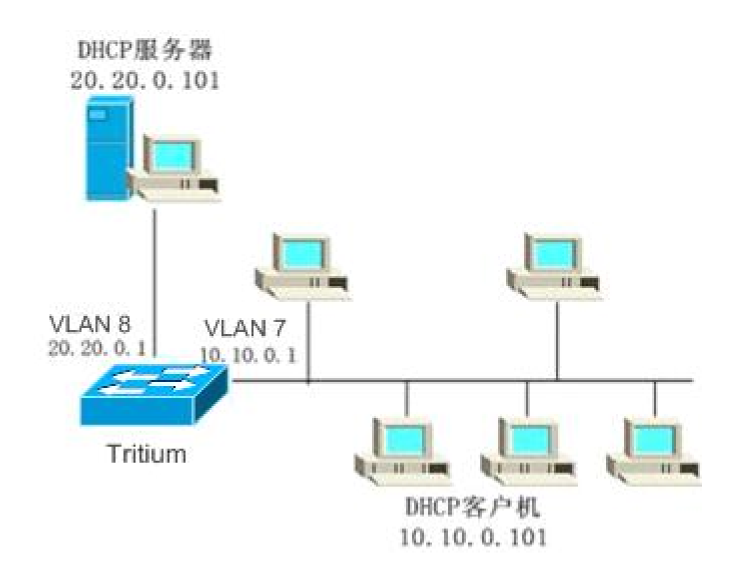


图 1-11 DHCP 中继组网图

三、配置步骤

配置说明：交换机的 VLAN 已经创建好。

配置交换机：

！配置交换机 IP 地址和地址掩码以使其和 DHCP 客户机和服务器分属于同一个网段

Tritium(config)# interface vlan *8*

Tritium(config-vlan-if)# ip address *20.20.0.1 255.255.0.0*

Tritium(config-vlan-if)# exit

Tritium(config)# interface vlan *7*

Tritium(config-vlan-if)# ip address *10.10.0.1 255.255.0.0*

Tritium(config-vlan-if)# exit

！配置 IP 中继地址以指明 DHCP 服务器的位置

Tritium(config)# ip dhcp-server *20.20.0.101*

DHCP 服务器的配置略。

1.4.4.2. 交换机充当 DHCP 服务器

一、组网需求 DHCP 客户机所在的网段为 10.10.0.0，交换机充当 DHCP 服务器，配置：

- 客户端所在网段为 10.10.0.0。排除地址范围为 10.10.0.9～10.10.0.25，10.10.0.100。
- 配置 MAC 地址 01:b7:08:13:88:11 的客户主机绑定 IP 地址 10.10.0.55；
- 配置缺省交换机为 10.10.0.100
- 配置 DNS 服务器地址为 20.20.0.101；
- 配置域名为 forward.com；
- 配置地址租用时间为 2 天。

二、组网图

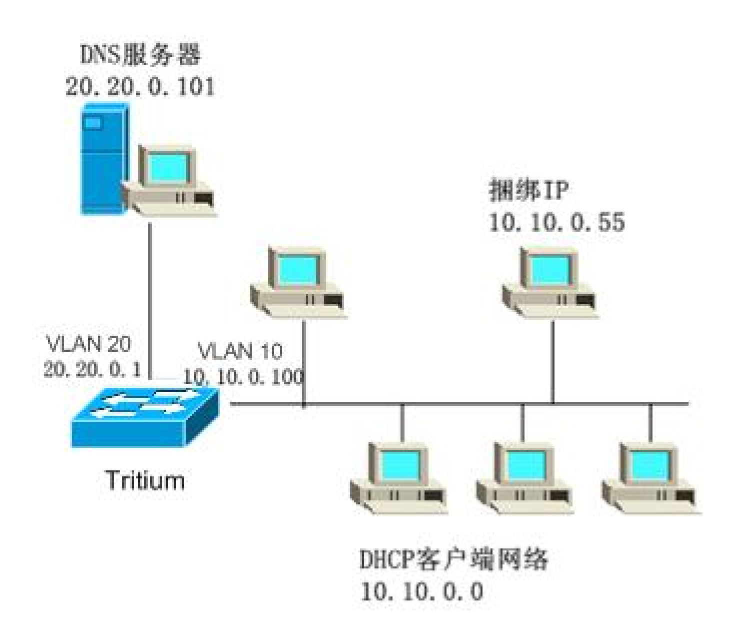


图 1-12 交换机充当 DHCP 服务器组网图

三、配置步骤

配置说明：交换机的 VLAN 已经创建好。

配置交换机：

！配置交换机 IP 地址和地址掩码以使其和 DHCP 客户机分于同一个网段

Tritium(config)# interface vlan *10*

Tritium(config-vlan-if)# ip address *10.10.0.100 255.255.0.0*

Tritium(config-vlan-if)# exit

！配置 IP 地址池

Tritium(config)# ip dhcp pool *test1*

Tritium(dhcp-config)# network *10.10.0.0 255.255.0.0*

Tritium(dhcp-config)# exit

Tritium(config)# ip dhcp excluded-address*10.10.0.9 10.10.0.25*

Tritium(config)# ip dhcp excluded-address*10.10.0.100*

！配置 DNS 服务器地址、域名及缺省交换机地址

Tritium(dhcp-config)# default-router *10.10.0.100*

Tritium(dhcp-config)# dns-server *20.20.0.101*

Tritium(dhcp-config)# domain-name *forward.com*

！配置 IP 地址租用时间为 2 天

Tritium(dhcp-config)# lease *2*

！配置 IP 地址绑定

Tritium(config)# ip dhcp pool *test2*

Tritium(dhcp-config)# hardware *01b7.0813.8811*

Tritium(dhcp-config)# host *10.10.0.55 255.255.0.0*

Tritium(dhcp-config)# default-router *10.10.0.100*

Tritium(dhcp-config)# dns-server *20.20.0.101*

Tritium(dhcp-config)# domain-name *forward.com*

Tritium(dhcp-config)# lease *2*

1.5. DNS

1.5.1.DNS 简介

在整个 Internet 上，成千上万台主机都是通过 IP 地址来区分的。当大家对 IP 地址了解后，发现去记这些数字很难记。在这种需求之下，我们有了域名系统 DNS（Domain Name System），一种字符串形式的主机命名机制。

域名系统（DNS）是一种分布式网络目录服务，使用一种有层次的命名方式，为网上的设备指定一个有意义的名字，并且在网络上设置域名解析服务器，建立域名与 IP 地址的对应关系，采用客户服务器方式工作。这样一来用户就可以使用便于记忆的、有意义的域名，从

而用于域名与 IP 地址的相互转换，以及控制因特网的电子邮件的发送等通信。

DNS 代理指 DNS 客户端不直接与 DNS 服务器端通信，而是把请求信息发送给中间代理，由代理完成与服务器端的交互并把结果返回给客户端。DNS 代理需要客户端主机指定 DNS 服务器地址为代理三层交换机，而 DNS 透明代理则不需要修改客户端主机 DNS 服务器配置。

1.5.1.1. DNS 命名

DNS 不是随便命名的，在 DNS 命名方式中，采用了分散和分层的机制来实现域名空间的委派授权以及域名与地址相转换的授权。通过使用 DNS 的命名方式来为遍布全球的网络设备分配域名，而这则是由分散在世界各地的服务器实现的。

理论上，DNS 协议中的域名标准阐述了一种可用任意标签值的分布式的抽象域名空间。任何组织都可以建立域名系统，为其所有分布结构选择标签，但大多数 DNS 协议用户遵循官方因特网域名系统使用的分级标签。常见的顶级域是：COM、EDU、GOV、NET、ORG、BIZ，另外还有一些带国家代码的顶级域。

DNS 的分布式机制支持有效且可靠的名字到 IP 地址的映射。多数名字可以在本地映射，不同站点的服务器相互合作能够解决大网络的名字与 IP 地址的映射问题。单个服务器的故障不会影响 DNS 的正确操作。DNS 是一种通用协议，它并不仅限于网络设备名称。

1.5.1.2. 静态 DNS

域名解析有两种方式，一种是查询本地静态域名解析表，另外一种是直接向 DNS 服务器查询，也称为动态查询。两种 DNS 解析模式相辅相成，首先进行 DNS 的静态解析，如果没有找到相应的 IP 地址，然后再进行动态的 DNS 解析，对于一些常用的 DNS 对应关系，最后写入静态 DNS 域名解析表中，可以提高域名解析的速度。

静态域名解析是通过静态域名解析表进行的，即手动建立域名和 IP 地址之间的对应关系表，该表的作用类似于 Windows 9X 操作系统下的 hosts 文件。当客户机需要域名所对应的 IP 地址时，即到静态域名解析表中去查找指定的域名，从而获得所对应的 IP 地址。

1.5.1.3. 动态 DNS

动态解析有专用的域名解析服务器，负责接受客户提出的域名解析请求。服务器首先在本机数据库内部解析，如果判断不属于本域范围之内，就将请求交给上一级的域名解析服务

器，直到完成解析。解析的结果或者为 IP 地址，或者域名不存在，并将解析的结果反馈给客户机。

用户程序对域名服务器（DNS Server）的访问是通过 DNS 客户端（DNS Client）的一个地址解析器（Resolver）来完成的。工作过程如下图所示：用户程序首先向 DNS Client 发出请求，DNS Client 收到请求后，首先查询本机数据库/缓存，如果没有发现所要查找的映射项，就向域名服务器发送查询报文，收到响应后再解析域名服务器发回来的响应报文，并根据响应报文的内容决定下一步的操作。

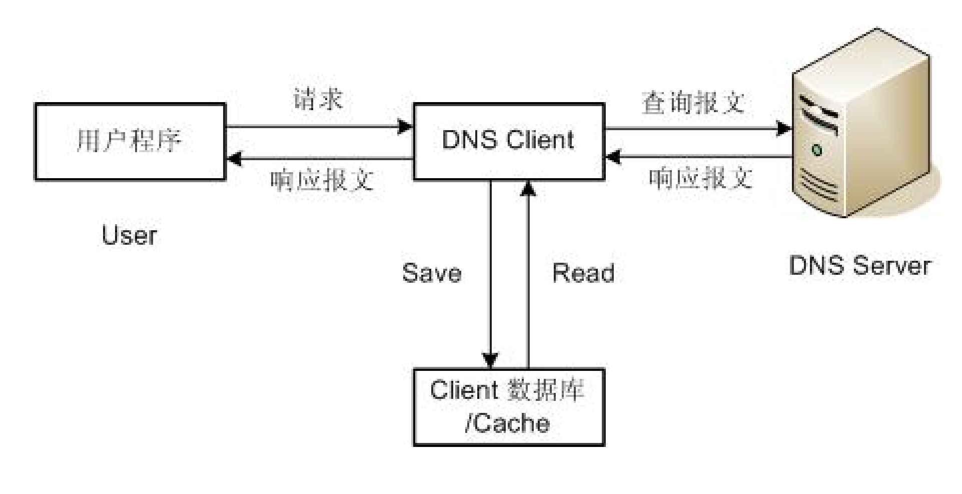


图 1-13 动态 DNS

用户程序、解析器和域名服务器以及解析器上的缓冲区关系如上图所示，其中解析器和缓冲区集成在一起构成 DNS Client，它的作用是接受用户程序的 DNS 咨询，并对其做出反应。一般来说，用户程序和解析器是在同一台主机上，域名服务器可以和它们在同一台主机上，也可以在不同的主机上，一般情况下是在不同的主机上。

动态域名解析支持缓存功能，对于每次动态解析成功的域名 IP 地址映射，存放在内存的动态域名缓存区中，下一次查询相同域名的时候，就可以直接从缓存区中读取，不用向域名服务器请求了。缓存区中的映射在一段时间后被老化删除，保证能够及时从域名服务器得到最新的内容。老化时间由域名服务器设置，三层交换机从协议报文中获得。

动态域名解析支持域名后缀列表功能，用户可以预先设置一些域名后缀，在域名解析的时候，用户只需要输入域名的部分字段，系统会自动将输入域名加上不同的后缀进行解析。举例说明，用户想查询域名“galaxywind.com”，那么可以在后缀列表中配置 com，然后输入“galaxywind”，系统会自动将输入域名与后缀连接成“galaxywind.com”进行查询。

1.5.2.DNS 配置

1.5.2.1. 开启 DNS 代理

表 1–15 开启 DNS 代理

命令	命令模式	功能说明
ip dns server	全局配置模式	使能 DNS 代理。
no ip dns server	全局配置模式	关闭 DNS 代理，缺省情况为关闭 DNS 服务。
ip dns transparent-proxy	全局配置模式	开启 DNS 透明代理服务
ip dns transparent-proxy enable	接口配置模式	在指定接口使能 DNS 透明代理功能，从该接口收到的 DNS 请求将被重定向到三层交换机的主控进行处理。

1.5.2.2. 配置 DNS 服务器

添加 DNS 服务器，只有配置好 DNS 服务器时 DNS 客户端和 DNS 代理才能正常工作；DNS 查询时，将按照配置顺序进行查询，只有当前面查询不到时方查询后面服务器，因此请严格按照 DNS 服务器的顺序进行配置。系统最多支持 6 个 DNS 服务器。

必须先配置 DNS 服务器，才能使能 DNS 客户端服务。缺省情况下，没有 DNS 服务器。

表 1–16 配置 DNS 服务器

命令	命令模式	功能说明
ip name-server *server* [*server*]…	全局配置模式	添加 DNS 服务器。
no ip name-server [*server*]…	全局配置模式	删除 DNS 服务器。

1.5.2.3. 开启 DNS 客户端服务

使能 DNS 客户端服务，使能前必须先配置 DNS 服务器。

表 1–17 开启 DNS 客户端服务

命令	命令模式	功能说明
ip domain lookup	全局配置模式	使能 DNS 客户端服务。
no ip domain lookup	全局配置模式	关闭 DNS 客户端服务，缺省情况为关闭状态。

1.5.2.4. 添加 DNS 默认域名

添加 DNS 默认域名，当查询名字没有找到时，添加默认域名到尾部进行查询。本配置是可选项，系统最多支持 6 个默认域名。

缺省情况下，没有默认域名。

表 1–18 添加 DNS 默认域名

命令	命令模式	功能说明
ip domain name *string* [*string*]…	全局配置模式	添加 DNS 默认域名。
no ip domain name [*string*]…	全局配置模式	删除 DNS 默认域名。

1.5.2.5. 配置 DNS 查询重试次数

配置 DNS 重试次数，当 DNS 请求发送超时后，客户端将重发该报文，当重试次数超过配置值后仍未收到响应时，将返回失败。

缺省的重试次数为 2 次。

表 1–19 配置 DNS 查询重试次数

命令	命令模式	功能说明
ip domain retry *number*	全局配置模式	配置 DNS 查询重试次数。
no ip domain retry	全局配置模式	恢复 DNS 查询重试次数的缺省值。

1.5.2.6. 配置 DNS 查询超时时间

配置 DNS 超时时间，即 DNS 请求报文发送后等待服务器端响应的时间。

缺省的超时时间为 3 秒。

表 1–20 配置 DNS 查询超时时间

命令	命令模式	功能说明
ip domain timeout *seconds*	全局配置模式	配置 DNS 查询超时时间。
no ip domain timeout	全局配置模式	恢复 DNS 查询超时时间的缺省值。

1.5.2.7. DNS 查看

通过 **show** 命令可以对 DNS 配置后的情况进行查看。

表 1–21 DNS 查看

命令	命令模式	功能说明
show ip dns config	特权用户模式	显示 DNS 配置信息，主要包括超时时间，重试次数，服务器配置，默认域名配置。
show ip dns statistics	特权用户模式	显示 DNS 统计信息，主要包括接收查询个数，回复个数，丢弃个数。

1.5.3.DNS 配置示例

1.5.3.1. 三层交换机客户端功能配置



图 1-14 DNS 组网图

功能描述：如上图所示，DNS 服务器 IP 地址为 20.1.1.2/16，Tritium Router 作为 DNS client，能通过 galaxywind.com 域名访问 10.1.1.2/16，并且能解析 Internet 上的其他 DNS 域名。。

基本条件：网络中有能够正常工作的 DNS 服务器。如果需要访问本局域网静态域名，还需要在该 DNS 服务器上配置静态域名映射。

Tritium Switch 配置步骤：

! 配置 Tritium Switch 接口 IP 地址

Tritium(config)# interface fastethernet *1/1*

Tritium(config-if)# ip address *10.1.1.1 255.255.0.0*

Tritium(config-if)# exit

Tritium(config)# interface fastethernet *1/2*

Tritium(config-if)# ip address *20.1.1.1 255.255.0.0*

Tritium(config-if)# exit

！配置 DNS 服务器

Tritium(config)# ip name-server *20.1.1.2*

！开启 DNS 客户端服务

Tritium(config)# ip domain lookup

！添加 DNS 默认域名

Tritium(config)# ip domain name *com net*

1.5.3.2. DNS 服务器配置示例

功能描述：三层交换机作为本局域网的 DNS 服务器，提供域名解析服务。

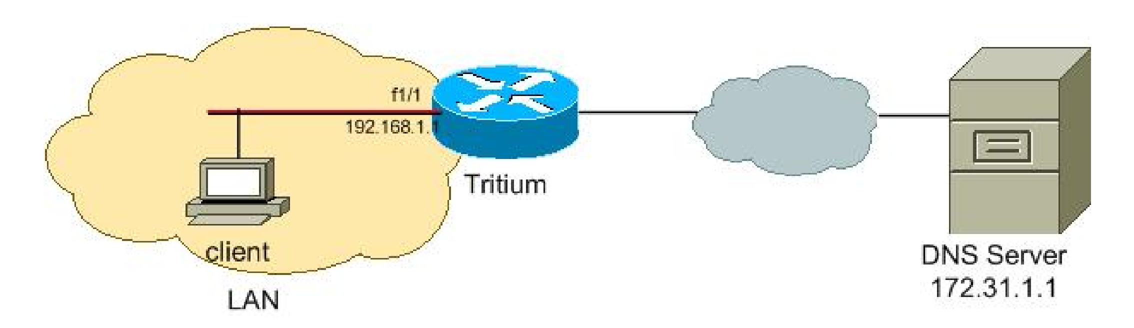


图 **1-15 DNS** 服务器（代理）组网图

Tritium Router 配置步骤：

！配置 Tritium Router 接口 IP 地址

Tritium(config)# interface fastethernet *1/1*

Tritium(config-if)# ip address *192.168.1.1 255.255.0.0*

Tritium(config-if)# exit

！配置 DNS 服务器

Tritium(config)# ip name-server *172.31.1.1*

！开启 DNS 服务器服务

Tritium(config)# ip dns server

如果不能动态学习到到 **DNS** 服务器（**172.31.1.1**）的路由，还需要添加静态路由。

本局域网的主机将 **DNS** 服务器地址填写为：**192.168.1.1**。

1.5.3.3. DNS 透明代理配置示例

功能描述：三层交换机为本局域网主机提供 DNS 服务。各主机不需要将 DNS 服务器填写为三层交换机地址，但是报文必须要经过三层交换机转发（一般情况是主机将网关地址填写为三层交换机地址）。三层交换机将截获所有 DNS 请求报文并做处理。

！配置 Tritium Router 接口 IP 地址

Tritium(config)# interface fastethernet *1/1*

Tritium(config-if)# ip address *192.168.1.1 255.255.0.0*

Tritium(config-if)# exit

！配置 DNS 服务器

Tritium(config)# ip name-server *172.31.1.1*

！全局模式开启 DNS 透明代理服务

Tritium(config)# ip dns transparent-proxy

！接口模式使能 DNS 透明代理服务

Tritium(config-if)# ip dns transparent-proxy enable

如果不能动态学习到到 DNS 服务器（172.31.1.1）的路由，还需要添加静态路由。

本局域网的主机将 DNS 服务器地址将不做限制。一般需要指定网关地址为三层交换机地址。

1.5.3.4. DNS 调试命令

功能描述：打开和关闭 DNS 相关调试信息。

！打开 Tritium Router DNS 调试信息

Tritium# debug ipcls debug-level dns

！关闭 Tritium Router DNS 调试信息

Tritium# no debug ipcls

1.6. IP 杂项配置

1.6.1.IP 调试

三层交换机管理员可通过 **show** 命令和 **debug** 命令来监控和维护 IP 性能。

表 1–22 IP 调试

命令	命令模式	功能说明
debug arp	特权用户模式	打开 ARP 调试信息开关。
debug ip packet [**detail**]	特权用户模式	打开 IP 调试信息开关。
debug ip tcp packet	特权用户模式	打开 TCP 调试信息开关。
debug tcp transactions	特权用户模式	打开 TCP 会话调试信息开关。
debug ip udp	特权用户模式	打开 UDP 调试信息开关。

【示例】IP 性能监控和维护。

例 1：显示 IP 层接口表信息。

Tritium# show interfaces fastethernet *2/1*

```
Fastethernet2/1 is administratively up,line protocol is up, link is up
MAC address is 06:01:02:02:00:02
Internet address is 200.26.12.161,netmask is 255.255.255.0
MTU is 1500,BW is 10000Kbit,Auto negotiation,Half duplex
Interface description :connected to shenzhen(192.168.0.99)
ARP type is ARPA, ARP Timeout:7200 seconds
ten seconds input rate:   0 bytes/sec, 1 packets/sec
ten seconds output rate: 0 bytes/sec, 0 packets/sec
0 unicast,0 multicast,45 broadcast packets input
5724 bytes input
0 unicast,0 multicast,1 broadcast packets output
64 bytes output4
```

上面显示信息表示：

- 快速以太网接口 2/1：处于正常工作状态（接口链路状态为 UP），IP 地址是 200.26.12.161，子网掩码长度为 24 位（255.255.255.0），最大传输单元是 1500 字节，接口说明：connected to shenzhen（192.168.0.99）。
- 快速以太网接口 2/1：接口收发报文的速率；接口收发报文总数；接口收

发报文的字节数。

例 2：打开 ARP 调试信息开关

Tritium# debug arp

Tritium# ping *192.168.0.111*

0x3EE4F60 (TELNETD): IP ARP : sent ARP request src 192.168.0.93:
  00a0.f71e.01c4, dst 192.168.0.224:?
0x6FBD140 (newLogLib): IP ARP : sent ARP request src 192.168.0.93:
  00a0.f71e.01c4, dst 192.168.0.177:?
0x4988DE0 (tNetTask): IP ARP: recvd rep src 192.168.0.224
  0003.4706.34b9, dst 192.168.0.93 00a0.f71e.01c4,interface
  FastEthernet8/0
0x4988DE0 (tNetTask): IP ARP: recvd rep src 192.168.0.177
  0005.5d0f.2f60, dst 192.168.0.93 00a0.f71e.01c4,interface
  FastEthernet8/0

上面显示信息表示：

- Tritium 三层交换机发送 ARP 请求报文，该请求报文格式为：源地址为本机接口地址 192.168.0.93，源 MAC 地址为 00a0.f71e.01c4，该 ARP 请求的报文查询 192.168.0.224 的 MAC 地址；
- Tritium 三层交换机发送 ARP 请求报文，该请求报文格式为：源地址为本机接口地址 192.168.0.93，源 MAC 地址为 00a0.f71e.01c4，该 ARP 请求的报文查询 192.168.0.177 的 MAC 地址；
- Tritium 三层交换机从快速以太网接口 8/0 收到来自 192.168.0.224 的主机发送的 ARP 应答报文，该应答报文中填写了 192.168.0. 224 的 MAC 地址 0003.4706.34b9；
- Tritium 三层交换机从快速以太网接口 8/0 收到来自 192.168.0.177 的主机发送的 ARP 应答报文，该应答报文中填写了 192.168.0. 177 的 MAC 地址 0005.5d0f.2f60。

例 3：打开 IP 调试信息开关，看 **ping** 的过程

Tritium# debug ip packet detail

Tritium# ping *192.168.0.177*

IP：s=192.168.0.88(local),d=192.168.0.177(FastEthernet0/0)
    len 48,send 3,TCP src=23,dst=1088

IP：s=192.168.0.177(FastEthernet0/0),d=192.168.0.88
len 20，rcvd 2,TCP src=1088，dst=23

上面显示信息表示：

- Tritium 三层交换机从快速以太网接口 0/0 发送 TCP 报文，该报文的源地址为本地接口 192.168.0.88（TCP 端口号为 23），目的地址为 192.168.0.177（TCP 端口号为 1088）；
- Tritium 三层交换机从快速以太网接口 0/0 接收到 TCP 报文，该报文的源地址为 192.168.0.177（TCP 端口号为 1088），目的地址为本地接口 192.168.0.88（TCP 端口号为 23）。

例 4：打开 TCP 调试信息开关

Tritium# debug ip tcp packet

```
0x3e1cbf8 (TBGPT): TCP: O SYNSENT 192.168.0.177: 179 192.168.0.88:
  1056
  seq 3881733053 DATA 0 ACK 0 SYN WIN 8192
0x4798a30 (tNetTask): TCP: I SYN_RCVD 192.168.0.177:
  1115192.168.0.88:
  23  seq 2745224011 ACK 0 WIN 16384
0x4798a30 (tNetTask): TCP: O SYNRCVD 192.168.0.177: 1115
  192.168.0.88:
  23  seq 3885637053 DATA 0 ACK 2745224011 SYN ACK WIN 8192
0x4798a30 (tNetTask): TCP: I ESTAB 192.168.0.177: 1115 192.168.0.88:
  23  seq 2745224011 ACK 3885637054 WIN 17520
```

上面显示信息表示：

1、打开 TCP 报文接收发送调试开关。

- 0：发送 TCP 报文；
- I：接收到 TCP 报文。

2、显示 TCP 连接状态。

- CLOSED：连接关闭；
- LISTEN：监听连接请求（被动打开 TCP 连接）；
- SYNSENT：已经发送 SYN（主动打开 TCP 连接）；
- SYNRCVD：已经发送并且接收 SYN，等待 ACK；

- ESTAB：连接建立（可以进行数据传输）；
- CLOSEWAIT：已经收到 FIN，等待应用程序关闭；
- FINWAIT1：已关闭，发送 FIN，等待 ACK 和 FIN；
- CLOSING：同时关闭，等待 ACK；
- LASTACK：收到的 FIN 已经关闭，等待 ACK；
- FINWAIT2：已经关闭，等待 FIN；
- TIMEWAIT：主动关闭后 2MSL 等待状态。

3、输入 TCP 报文解析。

- TCP 序列号（seq）：32 位的序列号由接收端计算机使用，重组分段的报文成最初形式。在动态路由网络中，一些报文很有可能使用不同的路由，因此，报文会乱序到达。这个序列号可以补偿传输中的不一致；
- TCP 应答号（ACK）：TCP 使用 32 位的应答（ACK）域标识下一个希望收到的报文的第一个字节。对一些没发生的事情作应答有点不直观，但收到 ACK 报文的源计算机会知道特定的段已经被收到。标识每个 ACK 的号是应答报文的序列号。该值只有在 ACK 标志被设置时才有效；
- 窗口大小（WIN）：目的主机使用 16 位的域告诉源主机，它想收到的每个 TCP 数据段大小；
- 输出 TCP 报文解析；
- TCP 序列号（seq）：32 位的序列号由接收端计算机使用，重组分段的报文成最初形式。在动态路由网络中，一些报文很有可能使用不同的路由，因此，报文会乱序到达。这个序列号可以补偿传输中的不一致；
- TCP 应答号（ACK）：TCP 使用 32 位的应答（ACK）域标识下一个希望收到的报文的第一个字节。对一些没发生的事情作应答有点不直观，但收到 ACK 报文的源计算机会知道特定的段已经被收到。标识每个 ACK 的号是应答报文的序列号。该值只有在 ACK 标志被设置时才有效；
- 窗口大小（WIN）：目的主机使用 16 位的域告诉源主机，它想收到的每个

TCP 数据段大小；

- 标志：TCP 报文中 6 位标志域，每 1 位标志可以打开一个控制功能，这六个标志是：紧急标志、有意义的应答标志、推、重置连接标志、同步序列号标志、完成发送数据标志。这些标志，以出现的先后顺序排列为 URG、ACK、PSH、RST、SYN 和 FIN；

- 发送缓存中比特数（DATA）。

例 5：打开 TCP 会话调试信息开关

Tritium# debug ip tcp transactions

0x4798a30 (tNetTask)：TCB06d8e7cc created
0x4798a30 (tNetTask)：TCB06d8e7cc bound to UNKNOWN.0
0x4798a30 (tNetTask)：state was LISTEN -> SYNRCVD [179 -> 192.168.0.177(1110)]
0x4798a30 (tNetTask)：Connection to 192.168.0.177：1110，received MSS 1460，MSS is 512
0x4798a30 (tNetTask)：sending SYN，seq 3711877053，ack 2577120891
0x4798a30 (tNetTask)：Connection to 192.168.0.177：1110，advertising MSS 1460
0x4798a30 (tNetTask)：state was SYNRCVD -> ESTAB [179 -> 192.168.0.177(1110)]
0x4798a30 (tNetTask)：TCB06d8e7cc setting property TCP_TOS (00)
0x4798a30 (tNetTask)：FIN processed
0x4798a30 (tNetTask)：state was ESTAB -> CLOSEWAIT [179 -> 192.168.0.177(1110)]
0x3e1cbf8 (TBGPT)：state was CLOSEWAIT -> LASTACK [179 -> 192.168.0.177(1110)]
0x3e1cbf8 (TBGPT)：sending FIN
0x4798a30 (tNetTask)：FIN acked
0x4798a30 (tNetTask)：state was LASTACK -> CLOSED [179 -> 192.168.0.177(1110)]
0x4798a30 (tNetTask)：TCB 0x06d8e7cc destroyed

上面显示信息表示：TCP 连接从侦听到正式创建连接到关闭连接的过程。

例 6：打开 UDP 调试信息开关

Tritium# debug ip udp

0x3e17bc0 (TELNETD)：UDP： sent src=192.168.0.88(520)，dst=224.0.0.9(520)，length=24
0x4798a30 (tNetTask)：UDP： rcvd src=192.168.0.88(520)，dst=224.0.0.9(520)，length=24

上面显示信息表示：

- 发送（sent）UDP 报文，该报文的源地址为本地接口 192.168.0.88（UDP 端口号为 520，是 RIP 报文保留端口号），目的地址为多播地址 224.0.0.9（224.0.0.9 为 RIP 保留多播地址，UDP 端口号为 520）；

- 接口（192.168.0.88）运行 RIP v2 路由协议，该接口加入了 224.0.0.9 的多播组，所以收到目的地址为多播地址 224.0.0.9 的 RIP 报文（Tritium 三层交换机收到该 RIP 多播报文，由于发送该报文的源地址为接口本身，在 RIP 模块处理中，实际上该报文被丢弃）。

第2章 IPv6 寻址和服务

2.1. 概述

互联网已经成为现代社会信息基础设施的重要组成部分，在国民经济发展和社会进步中起着举足轻重的作用，同时也成为当今高科技发展的重要支撑环境，互联网的巨大成功有目共睹。

现在被全球广泛使用的互联网协议 IPv4 是“互联网协议第四版”，已经有 30 年的历史。从技术上看，尽管 IPv4 在过去的应用具有辉煌的业绩，但是现在看来已经露出很多弊端。

全球范围内 WLAN、2.5G、3G 无线移动数据网络的发展加快了以互联网为核心的通信模式的形成，由于移动通信用户的增长要比固定网用户快得多，特别是各种具有联网功能的移动终端的迅猛发展，考虑到随时随地的、任何形式、直接的个人多媒体通信的需要，现有的 IPv4 已经远远不能满足网络市场对地址空间、端到端的 IP 连接、服务质量、网络安全和移动性能的要求。因此人们寄希望于新一代的 IP 协议来解决以上问题。

IPv6 协议正是基于这一思想提出的，它是“互联网协议第六版”的缩写。在设计 IPv6 时不仅仅扩充了 IPv4 的地址空间，而且对原 IPv4 协议各方面都进行了重新考虑，做了大量改进。除了提出庞大的地址数量外，IPv6 与 IPv4 相比，还有很多的工作正在进行以期得到更高的安全性、更好的可管理性，对 QoS 和多播技术的支持也更为良好。

2.2. IPv6 地址配置

2.2.1.IPv6 地址介绍

2.2.1.1. IPv4 和 IPv6 报头格式主要区别

IPv6 报头采用基本报头+扩展报头链组成的形式，这种设计可以更方便地增添选项以达到改善网络性能、增强安全性或添加新功能的目的。

2.2.1.2. IPv6 报头格式

固定的 IPv6 基本报头

IPv6 基本报头被固定为 40bytes，使路由器可以加快对数据包的处理速度，提高了转发效率，从而提高网络的整体吞吐量，使信息传输更加快速。

<table>
<tr><td colspan="4">ipv6 报头格式</td></tr>
<tr><td>4bit 版本号</td><td>4bit 优先级</td><td colspan="2">24bit 流标签</td></tr>
<tr><td colspan="2">净荷长度(16bit)</td><td>下一报头(8bit)</td><td>HOP 限制(8bit)</td></tr>
<tr><td colspan="4">源 IP 地址（128bit）</td></tr>
<tr><td colspan="4">目的 IP 地址（128bit）</td></tr>
<tr><td colspan="4">……</td></tr>
</table>

简化的 IPv6 基本报头

IPv6 基本报头中去掉了 IPv4 报头中阴影部分的字段，其中段偏移和选项和填充字段被放到 IPv6 扩展报头中进行处理。

去掉报头校验(HeaderChecksum)，中间路由器不再进行数据包校验，去掉此字段的原因有三：一是因为大部分二层链路层已经对数据包进行了校验和纠错控制，链路层的可靠保证使得三层网络层不必再进行报头校验；二是端到端的四层传输层协议也有校验功能以发现错包；三是报头校验需随着 TTL 值的变化在每一跳重新进行计算，增加包传送的时延。

IPv6 基本报头中去掉与 IP 分片相关的域，使得路由器无需再对数据包进行分片，而分片工作由源终端设备根据最大传输单元 MTU 路径发现来进行。这样 IPv6 的数据包可以远远超过 64kbit/s，应用程序可以利用 MTU，获得更快、更可靠的数据传输。

IPv6 报头新增流标记字段

IPv6 协议不仅保存了 IPv4 报头中的业务类别字段，而且新增了流标记字段，使得业务可以根据不同的数据流进行更细的分类，实现优先级控制和 QoS 保障，极大地改善了 IPv6 的服务质量。

IPv6 报头采用 128bit 地址长度

这是 IPv4 与 IPv6 最主要的区别。IPv4 采用 32bit 长度，理论上可以提供大约 43 亿个

IP 地址，这么多的 IP 地址似乎可以满足网络连接的需要，但事实上网络中任意交换机和交换机任意端口均需一个独立地址，为此网络缺乏足够地址满足各种潜在的用户。

IPv6 采用 128bit 长度，相对 IPv4，增加了 296 倍的地址空间。按保守方法估算 IPv6 实际可分配的地址，整个地球的每平方米面积上仍可分配 1000 多个地址。这样几乎可以不受限制地提供 IP 地址，从而确保了端到端连接的可能性。

IPv4 地址表示为点分十进制格式，32 位的地址分成 4 个 8 位分组，每个 8 位写成十进制，中间用点号分隔。而 IPv6 的 128 位地址则是以 16 位为一分组，每个 16 位分组写成 4 个十六进制数，中间用冒号分隔，称为冒号分十六进制格式。例如:21DA:00D3:0000:2F3B:02AA:00FF:FE28:9C5A 是一个完整的 IPv6 地址。

IPv6 的地址表示

IPv6 地址中每个 16 位分组中的前导零位可以去除做简化表示，但每个分组必须至少保留一位数字。如上例中的地址，去除前导零位后可写成:21DA:D3:0:2F3B:2AA:FF:FE28:9C5A。

某些地址中可能包含很长的零序列，为进一步简化表示法，还可以将冒号十六进制格式中相邻的连续零位合并，用双冒号"::"表示。"::"符号在一个地址中只能出现一次，该符号也能用来压缩地址中前部和尾部的相邻的连续零位。例如地址 1080:0:0:0:8:800:200C:417A，0:0:0:0:0:0:0:1，0:0:0:0:0:0:0:0 分别可表示为压缩格式 1080::8:800:200C:417A，::1，:: 。

在 IPv4 和 IPv6 混合环境中，有时更适合于采用另一种表示形式：x:x:x:x:x:x:d.d.d.d，其中 x 是地址中 6 个高阶 16 位分组的十六进制值，d 是地址中 4 个低阶 8 位分组的十进制值（标准 IPv4 表示）。例如地址 0:0:0:0:0:0:13.1.68.3 ，0:0:0:0:0:FFFF:129.144.52.38 写成压缩形式为::13.1.68.3，::FFFF.129.144.52.38 。

要在一个 URL 中使用文本 IPv6 地址，文本地址应该用符号"["和"]"来封闭。例如文本 IPv6 地址 FEDC:BA98:7654:3210:FEDC:BA98:7654:3210 写作 URL 示例为 http://[FEDC:BA98:7654:3210:FEDC:BA98:7654:3210]:80/index.html。

2.2.1.3. IPv6 地址类型

所有类型的 IPv6 地址都被分配到接口，而不是节点。IPv6 地址是单个或一组接口的 128 位标识符，有三种类型：

(1) 单播（Unicast）地址

单一接口的标识符。发往单播地址的包被送给该地址标识的接口。对于有多个接口的节点，它的任何一个单播地址都可以用作该节点的标识符。IPv6 单播地址是用连续的位掩码聚集的地址，类似于 CIDR 的 IPv4 地址。IPv6 中的单播地址分配有多种形式，包括全部可聚集全球单播地址、NSAP 地址、IPX 分级地址、链路本地地址以及运行 IPv4 的主机地址。单播地址中有下列两种特殊地址：

- 不确定地址

单播地址 0:0:0:0:0:0:0:0 称为不确定地址。它不能分配给任何节点。它的一个应用示例是初始化主机时，在主机未取得自己的地址以前，可在它发送的任何 IPv6 包的源地址字段放上不确定地址。不确定地址不能在 IPv6 包中用作目的地址，也不能用在 IPv6 路由头中；

- 回环地址

单播地址 0:0:0:0:0:0:0:1 称为回环地址。节点用它来向自身发送 IPv6 包。它不能分配给任何物理接口。

(2) 任意播（AnyCast）地址

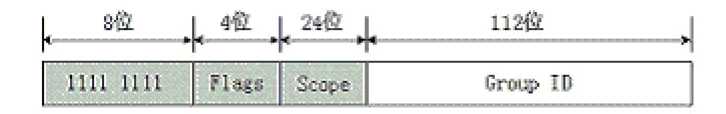
一组接口（一般属于不同节点）的标识符。发往任意播地址的包被送给该地址标识的接口之一（路由协议度量距离最近的）。IPv6 任意播地址存在下列限制：

· 任意播地址不能用作源地址，而只能作为目的地址；

· 任意播地址不能指定给 IPv6 主机，只能指定给 IPv6 路由器；

(3) 组播（MultiCast）地址

一组接口（一般属于不同节点）的标识符。发往多播地址的包被送给该地址标识的所有接口。地址开始的 11111111 标识该地址为组播地址。



IPv6 组播地址

IPv6 中没有广播地址，它的功能正在被组播地址所代替。另外，在 IPv6 中，任何全“0”和全“1”的字段都是合法值，除非特殊地排除在外的。特别是前缀可以包含“0”值字段或以“0”为终结。一个单接口可以指定任何类型的多个 IPv6 地址（单播、任意播、组播）或范围。

IPv6 的可聚集全球单播地址

IPv6 为点对点通信设计了一种具有分级结构的地址，这种地址被称为可聚集全球单播地址（Aggregatable Global Unicast Address），它在 RFC2374 中定义。可聚集地址具有三个层次的分级结构：

- 公用拓扑：提供公用互联网传送服务的供应商和交换局群体；
- 站点拓扑：本地的特定站点或组织，不提供到本站点以外节点的公用传送服务；
- 接口标识符：标识链路上的接口；

可聚集全球单播地址开始 3 个地址位是地址类型前缀，用于区别其它地址类型。其后的 13 位 TLA ID、32 位 NLA ID、16 位 SLA ID 和 64 位主机接口 ID，分别用于标识分级结构中自上向下排列的 TLA （Top Level Aggregator，顶级聚集体）、NLA（Next Level Aggregator，下级聚集体）、SLA（Site Level Aggregator，站点级聚集体）和主机接口。RES 保留，以备将来 TLA 或 NLA 扩充用。TLA 是与长途服务供应商和电话公司相互连接的公共网络接入点，它从国际 Internet 注册机构如 IANA 处获得地址。NLA 通常是大型 ISP，它从 TLA 处申请获得地址，并为 SLA 分配地址。SLA 也可称为订户（subscriber），它可以是一个机构或一个小型 ISP。SLA 负责为属于它的订户分配地址。SLA 通常为其订户分配由连续地址组成的地址块，以便这些机构可以建立自己的地址分级结构以识别不同的子网。分级结构的最底层是网络主机。

设计这样的地址格式是为了既支持基于当前供应商的聚集，又支持被称为交换局的新的聚集类型。其组合使高效的路由聚集可用于直接连接到供应商和连接到交换局两者的站点上。站点可以选择连接到两种类型中的任何一种聚集点。

2.2.1.4. 自动配置

IPv6 协议支持地址自动配置，这是一种即插即用的机制。IPv6 节点通过地址自动配置得到 IPv6 地址和网关地址。

IPv6 支持无状态地址自动配置和状态地址自动配置两种地址自动配置方式。在无状态地址自动配置方式下，需要配置地址的网络接口先使用邻居发现机制获得一个链路本地地址。网络接口得到这个链路本地地址之后，再接收路由器宣告的地址前缀，结合接口标识得到一个全球地址。而状态地址自动配置的方式，如动态主机配置协议（DHCP），需要一个 DHCP 服务器，通过客户机/服务器模式从 DHCP 服务器处得到地址配置的信息。

2.2.1.5. IPv6 路由器地址介绍

例如一台 IPv6 路由器可被分配以下几种单点传送地址：

- 每个接口的链路本地地址；
- 每个接口的单播地址（可以是一个和一个或多个可聚集全球地址）；
- 子网-路由器任意播地址；
- 其他任意播地址（可选）；
- 回环接口的回环地址（::1）。

同样，除以上这些地址外，路由器需要时刻保持收听以下多点传送地址上的信息流：

- 节点本地范围内的所有节点组播地址（FF01::1）；
- 节点本地范围内的所有路由器组播地址（FF01::2）；
- 链路本地范围内的所有节点组播地址（FF02::1）；
- 链路本地范围内的所有路由器组播地址（FF02::2）；
- 站点本地范围内的所有路由器组播地址（FF05::2）；
- 请求节点（solicited-node）组播地址（如果路由器的某个接口加入请求节点组）；
- 组播组组播地址（如果路由器的某个接口加入任何组播组）。

2.2.2.IPv6 协议控制开关

默认情况下 IPv6 功能将被关闭，用户需要通过命令配置才能打开 IPv6 功能。

如果用户需要在接口上使能 IPv6 功能，可以通过如下方式：

```
Tritium(config-vlan-if)#ipv6 enable          (1)
Tritium(config-vlan-if)#ipv6 address 2008::/64 eui-64    ( 2 )
Tritium(config-vlan-if)#ipv6 address fe80::11 link-local
Tritium(config-vlan-if)#ipv6 address autoconfig
```

当接口没有显式配置任何地址时，可以使用命令 1 启动接口 IPv6 功能。此时接口将自动计算出 link-local 地址，地址冲突检测成功后 IPv6 在接口上启动成功。如果地址冲突检测失败，link-local 地址将一直处于 DUP 状态，此后配置的其他地址也会处在 Tentative 状态。需要注意的是，如果接口显式配置有 IPv6 地址，则对应 no 命令不会关闭接口上 IPv6 功能。

另外接口上配置 IPv6 地址（命令 2）也会自动启动 IPv6 功能。

默认情况下，路由器不对 IPv6 报文进行转发，也不发送路由器通告报文。直到用户执

行如下命令：

```
Tritium(config)# ipv6 unicast-routing
```

2.2.3.接口配置单播

通常情况下，地址配置是路由器接口通过 IPv6 协议进行通讯的前提。因此 GF 路由器支持接口 ID 的计算，link-local 地址的自动配置和手动配置，全局地址的自动配置和手动配置。

默认条件下，接口的 link-local 地址由路由器自动配置。同时也支持手工配置 link-local 地址。为了和 Cisco 实现相兼容，路由器接口只允许配置一个 link-local 地址。如果 Link-local 地址未能配置成功（冲突状态），其他地址也不能正常配置。环回接口的 link-local 地址实际上是没有意义的，但是为了和其他机器一致，路由器需要为使能了 IPv6 的环回接口产生一个 link-local 地址。产生地址的 Interface ID 使用 FastEthernet0/0 的接口 ID。

路由器接口支持配置多个全局 IPv6 地址，而且配置的地址不需要通过 **Primary/Secondary** 关键字区别开来。同时，还支持指定前缀由路由器构造地址和直接指定地址两种方式。路由器也支持全局地址的无状态自动配置，此时路由器向链路上的其他路由器学习前缀信息并自动配置地址。

相关配置的命令罗列如下：

表 2–1 接口配置命令

命令	命令模式	功能说明
ipv6 address *ipv6-address link-local*	接口配置模式	配置接口的 IPv6 本地链路地址。
no ipv6 address *ipv6-address link-local*	接口配置模式	删除接口的 IPv6 本地链路地址。
ipv6 address *ipv6-address/prefix-length*	接口配置模式	配置接口的 IPv6 全局地址。
no ipv6 address *ipv6-address/prefix-length*	接口配置模式	删除接口的 IPv6 全局地址。
ipv6 address *ipv6-prefix/prefix-length* **eui-64**	接口配置模式	配置接口的 IPv6 EUI 地址。
no ipv6 address *ipv6-prefix/prefix-length*	接口配置模式	删除接口的 IPv6 EUI 地址。

eui-64		
ipv6 address autoconfig	接口配置模式	使能接口的 IPv6 地址自动配置
no ipv6 address autoconfig	接口配置模式	取消接口的 IPv6 地址自动配置

2.2.4.配置 IPv6 接口地址示例

例 1：配置当前接口 IPv6 功能，自动生成 link-local 地址

Tritium(config-vlan-if)# ipv6 enable

查看配置结果如下，

Tritium #　show ipv6 interface vlan 17

VLAN 17 is administratively up, line protocol is up

　IPv6 is enabled, link-local address is fe80::211:f7ff:fefa:8811

　No global unicast address is configured

　Joined group address(es):

　　ff02::16

　　ff02::d

　　ff02::1:fffa:8811

　　ff02::2

　　ff02::1

　　ff01::1

　MTU is 1500 bytes

　ND DAD is enabled, number of DAD attempts: 1

　Reference count 22, flags 0003

例 2：配置当前接口 IPv6 EUI 地址，指定前缀

Tritium(config-vlan-if)# ipv6 address 2001:1::/64 eui

查看配置结果如下，

Tritium #　show ipv6 interface vlan 17

VLAN 17 is administratively up, line protocol is up
  IPv6 is enabled, link-local address is fe80::211:f7ff:fefa:8811
  Global unicast address(es):
    2001:1::211:f7ff:fefa:8811, subnet is 2001:1::/64 [EUI]
  Joined group address(es):
    ff02::16
    ff02::d
    ff02::1:fffa:8811
    ff02::2
    ff02::1
    ff01::1
  MTU is 1500 bytes
  ND DAD is enabled, number of DAD attempts: 1
  Reference count 28, flags 0003

例 **3**：配置当前接口 **IPv6 link-local** 地址。

Tritium(config-vlan-if)# ipv6 address fe80::11 link-local

查看配置结果如下，

VLAN 17 is administratively up, line protocol is up
  IPv6 is enabled, link-local address is fe80::11
  Global unicast address(es):
    2001:1::11, subnet is 2001:1::/64 [EUI]
  Joined group address(es):
    ff02::1:ff00:11
    ff02::16
    ff02::d
    ff02::2
    ff02::1
    ff01::1
  MTU is 1500 bytes
  ND DAD is enabled, number of DAD attempts: 1
  Reference count 29, flags 0003

例 **4**：配置当前接口自动配置 **IPv6** 地址

Tritium(config-vlan-if)# no ipv6 address 2001:1::/64 eui

Tritium(config-vlan-if)# ipv6 address autoconfig

查看配置结果如下，

VLAN 17 is administratively up, line protocol is up
  IPv6 is enabled, link-local address is fe80::11
  Global unicast address(es):
    2001:1::211:f7ff:fefa:8811, subnet is 2001:1::/64 [PRE]

valid lifetime 2591996 preferred lifetime 604796
Joined group address(es):
ff02::1:fffa:8811
ff02::1:ff00:11
ff02::16
ff02::d
ff02::2
ff02::1
ff01::1
MTU is 1500 bytes
ND DAD is enabled, number of DAD attempts: 1
Reference count 33, flags 0103

例 5：配置当前接口 IPv6 地址，同时配置多个 ipv6 地址

Tritium(config-vlan-if)# ipv6 address 2008::1/64

查看配置结果如下，

VLAN 17 is administratively up, line protocol is up

IPv6 is enabled, link-local address is fe80::11

Global unicast address(es):

2001:1::211:f7ff:fefa:8811, subnet is 2001:1::/64 [PRE]

valid lifetime 2591945 preferred lifetime 604745

2008::1, subnet is 2008::/64

Joined group address(es):

ff02::1:ff00:1

ff02::1:fffa:8811

ff02::1:ff00:11

ff02::16

ff02::d

ff02::2

ff02::1

ff01::1

MTU is 1500 bytes

ND DAD is enabled, number of DAD attempts: 1

Reference count 39, flags 0103

2.2.5.IPv6 地址配置排错

路由器是网络互连设备，因而在给接口配置 IPv6 地址时，我们必须明白组网需求和子网的划分，一般遵循如下原则：

- 成功配置：在需要配置接口的接口配置模式下成功配置 IPv6 地址；
- 链路状态指示正确：接口的管理状态和链路状态都应为 UP；
- 以太网口主 IPv6 地址必须与该以太网口所连的局域网在同一网段；

广域网两端的路由器的接口的 IPv6 地址必须在同一网段。

故障之一：从路由器 ping ipv6 局域网中某一主机不通。

故障排除：

首先检查该以太网口和局域网中主机的 IPv6 地址配置是否位于同一网段；

如果配置正确就可打开 debug ipv6 nd 调试开关，查看路由器是否正确地发送和接收 ns 和 na 报文，如果只有发送 ns 没有接收到 na 报文，则以太网物理层可能有问题。

故障之二：配置接口自动生成 IPv6 地址后未见相应 IPv6 地址生成。

故障排除：

打开 debug ipv6 nd 调试开关，查看路由器是否正确地发送和接收 rs 和 ra 报文，如果只有发送 rs 没有接收到 ra 报文，则检查网络中是否有使能了 ipv6 unicast-routing 的路由器存在，如果存在使能了 ipv6 unicast-routing 的路由器还不能收到 ra 报文则以太网物理层可能有问题。

2.3. 邻居管理配置

2.3.1.IPv6 邻居发现协议

IPv6 定义了邻居发现协议（Neighbor Discovery protocol，NDP），它使用一系列 IPv6 控制信息报文（ICMPv6）来实现相邻节点（同一链路上的节点）的交互管理，并在一个子网中保持网络层地址和链路层地址之间的映射。邻居发现协议中定义了 5 种类型的信息：

路由器通告、路由器请求、路由重定向、邻居请求和邻居通告。通过这些信息，实现了对以下功能的支持：

- 路由器发现：即帮助主机来识别本地路由器；
- 前缀发现：节点使用此机制来确定指明链路本地地址的地址前缀以及必须发送给路由器转发的地址前缀；
- 参数发现：帮助节点确定诸如本地链路 MTU 之类的信息；
- 地址自动配置：用于 IPv6 节点自动配置；
- 地址解析：替代了 ARP 和 RARP，帮助节点从目的 IP 地址中确定本地节点（即邻居）的链路层地址；
- 下一跳确定：可用于确定包的下一个目的地，即可确定包的目的地是否在本地链路上。如果在本地链路，下一跳就是目的地；
- 邻居不可达检测：帮助节点确定邻居（目的节点或路由器）是否可达；
- 重复地址检测：帮助节点确定它想使用的地址在本地链路上是否已被占用；
- 重定向：有时节点选择的转发路由器对于待转发的包而言并非最佳。这种情况下，该转发路由器可以对节点进行重定向，使它将包发送给更佳的路由器。例如，节点将发往 Internet 的包发送给为节点所在的内部网服务的默认路由器，该内部网路由器可以对节点进行重定向，以使其将包发送给连接在同一本地链路上的 Internet 路由器。

2.3.2.NS/NA 报文的收发

路由器必须能够正确收发 NS/NA 报文，能够正确处理报文中携带的选项。在此基础上，路由器需要实现地址解析，地址冲突检测和地址不可达检测功能。

2.3.3.RS/RA 报文的收发

路由器在使能 IPv6 转发功能后，需要向所在链路发送路由器通告，在通告中，需要携带各种选项。这些选项向用户提供了配置接口。

相关配置命令及其意义罗列如下：

ipv6 hop-limit *value*

设置 hop limit 的值，路由器发送报文设置的 hop limit，路由器将会把这个值通过 RA 通告到其他主机（默认为 64）

ipv6 nd advertisement-interval

设置 RA 中携带通告时间间隔选项

ipv6 nd dad attempts *value*

设置地址冲突检测的时间，此选项作为 RA 选项发布到链路上，默认情况下为 1 次

ipv6 nd ns-interval *milliseconds*

设置发送 NS 报文的间隔（地址冲突检测）

ipv6 nd ra-interval

ipv6 nd ra-lifetime

ipv6 nd reachable-time

用于配置 RA 中对应的参数。

ipv6 nd suppress-ra

在接口上关闭路由器通告功能。

通过 autoconfig 方式配置时，路由器还需发送 RS 和接收 RA 报文，并能正确进行处理。

2.3.4.邻居状态

（1）NONE：neighbor 创建时的初始状态，IP 报文发送时触发其进行地址解析，进入 INCOMPLETE 状态。

（2）INCOMPLETE：neighbor 正在进行地址解析，等待邻居请求回应。

（3）REACHABLE： neighbor 接收到邻居请求回应，进入 REACHABLE 状态。

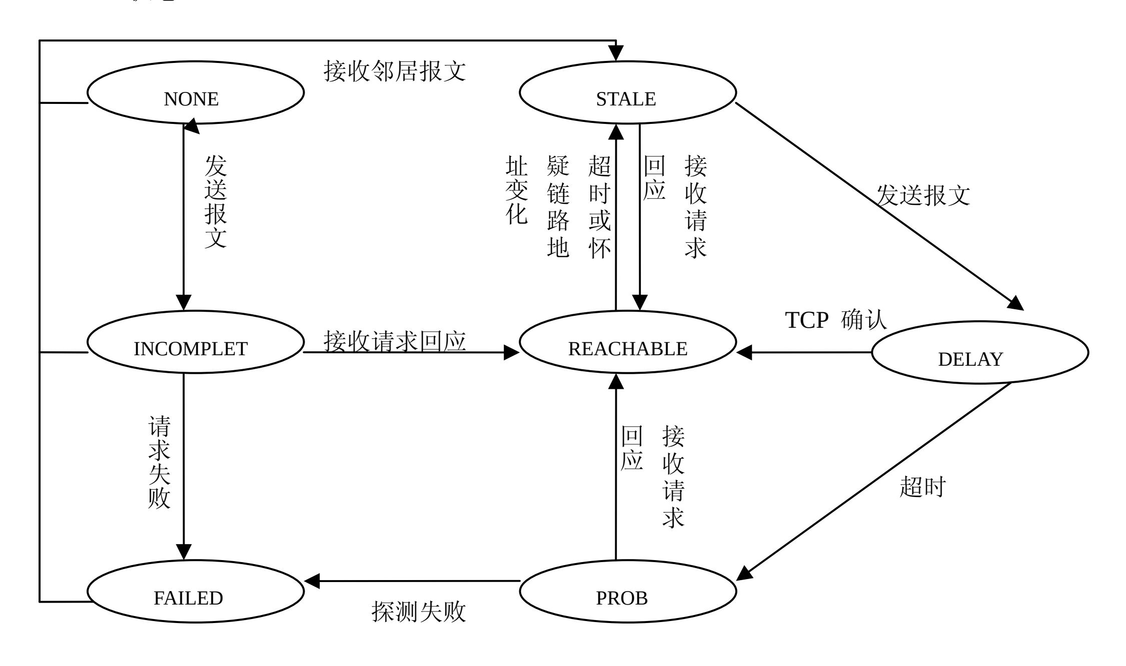
（4）STALE： neighbor 在 REACHABLE 状态经过 reachable_time，进入 STALE 状态；或者邻居报文更新了 neighbor 的链路地址。neighbor 在这个状态仍然有效，只不过在发送 IP 报文的同时状态变化为 DELAY，启动 neighbor 的可达性监测。

（5）DELAY：neighbor 在这个状态仍然有效。DELAY 状态为 neighbor 的可达性探测提供了一个优化，上层协议（比如 TCP）可以利用这一小段延迟时间确认邻居是否可达，如果上层协议发现可达（比如 TCP 收到 ack 报文），neighbor 状态变化为 REACHABLE。

（6）PROBE：neighbor 在 DELAY 状态经过小段延迟（delay_probe_time），没有上层协议确认 neighbor 可达，这时 neighbor 进入 PROBE 状态。neighbor 在这个状态仍然有

效，它将发送单播邻居请求，期待接收到请求回应。

（7）FAILED： neighbor 在 PROBE 状态或者 INCOMPLETE 状态多次请求无果后，进入 FAILED 状态。



邻居状态转换图

2.3.5.配置命令

表 2–2 IPv6 邻居管理配置命令列表

命令	命令模式	功能说明
clear ipv6 neighbors	特权用户模式	清空 IPv6 邻居表
ipv6 nd	接口配置模式	配置邻居管理相关参数
show ipv6 neighbors	特权用户模式	查看 IPv6 邻居表内容

例 1：清除 IPv6 邻居

Tritium# clear ipv6 neighbors

例 2：查看 IPv6 邻居

Tritium # show ipv6 neighbors
IPv6 Address Age Link-layer Addr State Interface
fe80::211:f7ff:fe09:90f 0 00:11:f7:09:09:0f DELAY VLAN 12

例 3：配置邻居管理相关参数

(1)设置 RA 中携带通告时间间隔选项。

Tritium(config-vlan-if)# ipv6 nd advertisment-interval

(2)配置 dad 检测重试次数 2。

Tritium(config-vlan-if)# ipv6 nd dad attempts 2

(3)配置 ns 间隔时长 3000 毫秒。

Tritium(config-vlan-if)# ipv6 nd ns-interval 3000

(4)配置 ra 间隔时长 6 秒。

Tritium(config-vlan-if)# ipv6 nd ra-interval 6

(5)配置可达性检测时长 **2000000 毫秒**。

Tritium(config-vlan-if)# ipv6 nd reachable-time 2000000

(6)配置接口上关闭路由器通告功能。

Tritium(config-vlan-if)# ipv6 nd suppress-ra

查看该接口上的配置结果,

```
Tritium # show running-config interface vlan 17
interface vlan 17
  ipv6 enable
  ipv6 address 2001:1::1/64
  ipv6 nd ns-interval 3000
  ipv6 nd reachable-time 2000000
  ipv6 nd ra-interval 6
  ipv6 nd advertisement-interval
  ipv6 nd suppress-ra
  ipv6 nd dad attempts 2
```

2.4. DHCPv6 配置

2.4.1.DHCPv6 概述

IPv6 协议提供了长达 128 位的庞大地址空间，如何实现地址高效合理的管理及分配成为必须解决的问题。IPv6 无状态地址配置协议[RFC2462]是当前广泛采用的 IPv6 地址自动配置方式。与主机相连的路由器开启 IPv6 路由器通告(RA)功能后，根据该通告内包含的网络地址前缀信息自动选择本机地址。这种方式需要手动为路由器配置 IPv6 地址，并不记录主机信息，同时也不提供除地址配置外的其他诸如 Domain Name，DNS 服务器等配置信息，仍然存在

一定缺陷。

DHCPv6 是 IPv6 中的动态主机配置协议，实现 IPv6 下配置的集中式管理，为主机动态分配全局的 IPv6 地址，通过响应主机的地址分配请求，来动态管理地址的租用和回收。

同 IPv4 DHCP 一样，DHCPv6 也分为两个部分：客户端和服务器。由客户端主动发起地址分配请求，服务器集中管理网络地址及配置信息，并响应客户端请求，客户端使用服务器传递的地址和信息进行工作。

由于 DHCPv6 的 Client 和 Server 通信中采用 linklocal 地址，只能在同一网络上进行，当客户端和服务器不在同一网络上，需要在中间节点上配置 DHCPv6 中继功能，实现跨网络的 DHCPv6 地址管理服务。

2.4.1.1. DHCPv6 的分配形式

DHCPv6 配置方式分为以下两种：

1）无状态 DHCPv6 服务

DHCPv6 服务器并不为主机提供地址分为服务，而是提供诸如 DNS 服务器等相关配置信息。

2）有状态地址配置

DHCPv6 服务器为主机提供全局地址分配及其他配置信息服务。

其中有状态地址配置方式根据分配地址类型的不同又分为两种情况

A）普通地址分配

服务器为请求者分配单个的地址，适用于为主机直接进行地址分配。

B）前缀代理（Prefix Delegation）

该功能是由 CISCO 提出并标准化，在层次结构的网络中，采用手工来进行地址分派，扩展性差且不易于管理，使用前缀代理后，下游的用户边缘网关（CPE）自动向上游 ISP 的 DHCPv6 服务器（SPE）提出地址前缀申请，获得地址分派后，下游的 CPE 进一步将该地址块划分为更小网络地址块分配给各个网络接口，并向主机提供网络地址分配。

2.4.1.2. DHCPv6 的工作原理

一次完整的普通 DHCPv6 地址分配会话包含以下过程：

- 地址分派请求

  DHCPv6 客户端向本地链路发送以 FF02::1:2（所有 DHCPv6 服务器及中继代理）为目的地址的分配请求（Solicit）。

- 服务器通告

  收到请求的服务器从本地地址池中分配一个可用地址，使用 Advertise 通告给客户端。

- 服务器选择

  客户端从收到的地址分配通告中采用一定策略选择一个服务器，并向该服务器发送 Confirm 请求确认分配通告的地址及其他配置信息。

- 服务器确认

  服务器收到后发送 Reply 回应，分配地址及传递配置信息。

除了以上分配方式外，DHCPv6 还支持快速分配方式，由客户端向服务器请求地址分配时，在选项中指定 rappid-commit，服务器收到后直接选择一个可用的地址，用 reply 报文响应给主机，主机收到后不再请求确认地址，而是默认该地址为有效地址，直接用作为本地配置进行工作。这种工作方式通常用于对实时性要求较高的网络中，用户要求快速获取到配置而不希望等待确认。

在快速配置模式中，应当注意以下问题：

- 在该网络中不得出现多个 DHCPv6 服务器，由于没有确认机制，因此如果存在多台服务器，可能出现地址分配冲突等非期望的结果，导致主机获取的地址配置不可用等异常情况；
- 必须是主机和服务器都具备快速协商能力，服务器上必须在接口指定地址池时配置 rappid-commit 参数，否则均按常规方式进行请求—通告—请求确认—确认过程。

2.4.2.DHCPv6 配置

2.4.2.1. 路由器充当 DHCPv6 服务器

Tritium 路由器支持 DHCPv6 服务器地址分配功能，配置如下:

表 2－3 配置 DHCPv6 服务器功能

命令	命令模式	功能说明
ipv6 dhcp pool *pool-name*	全局配置模式	启动 IPv6 DHCP 服务，配置 DHCP 地址池。
address prefix lifetime *life-time pref-time*	DHCPv6 配置模式	配置地址池可用地址前缀范围及有效时间。
ipv6 dhcp server	接口配置模式	配置接口使能 DHCPv6 Server 功能
ipv6 dhcp server *pool-name*	接口配置模式	配置接口应用的 DHCPv6 地址池范围

DHCPv6 管理及监控：

表 2－4 DHCPv6 管理及监控

命令	命令模式	功能说明
debug ipv6 dhcp	特权用户模式	打开 DHCPv6 调试信息。
debug ipv6 dhcp detail	特权用户模式	打开 DHCPv6 详细调试信息。
show ipv6 dhcp	特权用户模式	显示 DHCPv6 基本信息
show ipv6 dhcp binding {*client-address*}	特权用户模式	显示 DHCPv6 地址分配信息
show ipv6 dhcp interface	特权用户模式	显示使能 DHCPv6 功能的接口信息
show ipv6 dhcp pool	特权用户模式	显示 DHCPv6 地址池配置信息
clear ipv6 dhcp binding {**all** | *client-addr*}	特权用户模式	清除租约，回收地址

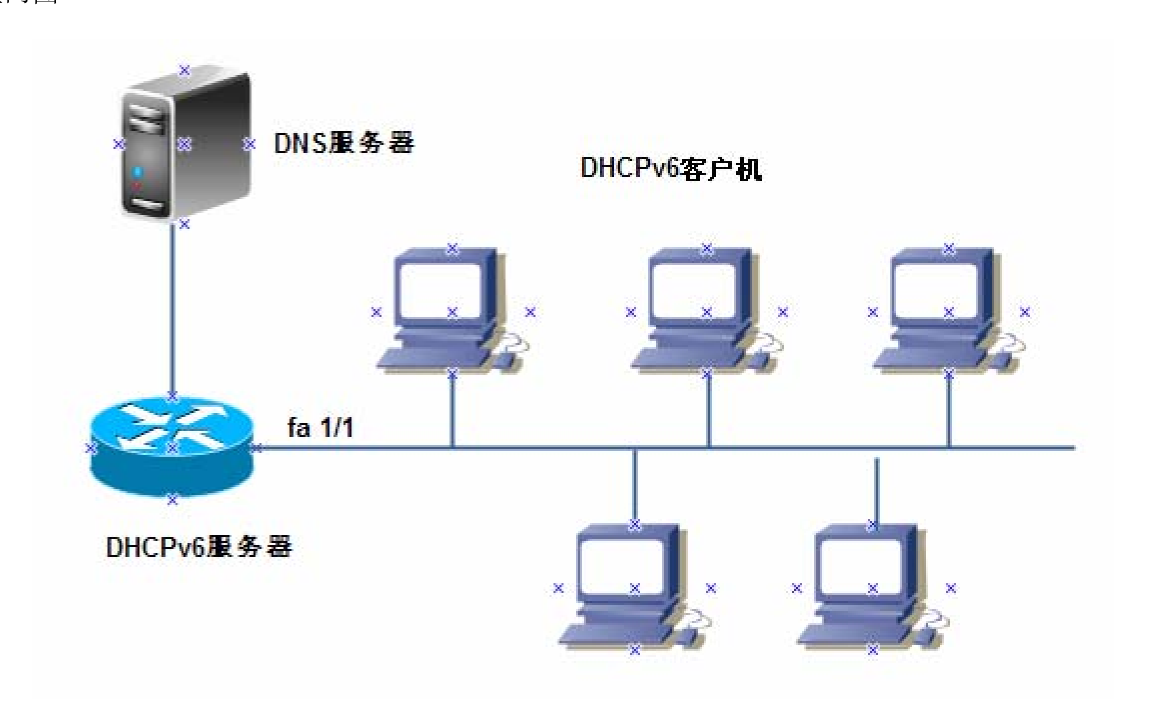
2.4.3.DHCPv6 配置实例

2.4.3.1. 路由器充当 DHCPv6 服务器

一、组网需求

- 客户机与 DHCPv6 服务器的 fa 1/1 相连；
- DNS 服务器为 23::3
- 路由器充当 DHCPv6 服务器，可用地址池为 44::/112
- 地址租期为 1 天，地址生存时间为 2 天；

二、组网图



三、配置步骤

！配置路由器使能 DHCPv6，设置 DNS 服务器、地址池范围及租期时间

Tritium(config)# ipv6 dhcp pool *usr1*

Tritium(config-dhcpv6)# address prefix *44::/112* **lifetime** *172800 86400 255.255*

Tritium(config-dhcpv6)# dns-server *23::3*

Tritium(config-dhcpv6# exit

！配置接口 fa 1/1 使能 IPv6 及 DHCPv6 Server 功能

Tritium(config)# interface fastethernet *1/1*

Tritium(config-if)# ipv6 dhcp server

Tritium(config-if)# ipv6 dhcp server *usr1*